Optimized Hyperspectral Imagery Anomaly Detection through Robust Parameter Design

DISSERTATION

Francis M. Mindrup, Major, USAF

AFIT/DS/ENS/11-04

DEPARTMENT OF THE AIR FORCE
AIR UNIVERSITY

AIR FORCE INSTITUTE OF TECHNOLOGY

Wright-Patterson Air Force Base, Ohio

AFIT/DS/ENS/11-04

OPTIMIZED HYPERSPECTRAL IMAGERY ANOMALY DETECTION THROUGH ROBUST PARAMETER DESIGN

DISSERTATION

Presented to the Faculty

Graduate School of Engineering and Management

Air Force Institute of Technology

Air University

Air Education and Training Command

in Partial Fulfillment of the Requirements for the

Degree of Doctor of Philosophy

Francis M. Mindrup, B.S., M.S.

Major, USAF

October 2011

AFIT/DS/ENS/11-04

OPTIMIZED HYPERSPECTRAL IMAGERY ANOMALY DETECTION THROUGH ROBUST PARAMETER DESIGN

Francis M. Mindrup, B.S., M.S.
Major, USAF

Approved:

Dr. Kenneth W. Bauer Dissertation Advisor	Date
Dr. J. O. Miller Committee Member	Date
Dr. Mark E. Oxley Committee Member	Date
Maj Mark A. Friend, PhD Committee Member	Date

Accepted:

M. U. Thomas Dean, Graduate School of Engineering and Management	Date

For my loving wife, daughter and sons

AFIT/DS/ENS/11-04

Abstract

Advances in sensor technology necessitate fast and accurate methods to deal with an ever growing wellspring of information. Anomaly detection algorithms for hyperspectral imagery (HSI) are an important first step in the analysis chain which can reduce the overall amount of data to be processed. The actual amount of data reduced depends greatly on the accuracy of the anomaly detection algorithm implemented. Most, if not all, anomaly detection algorithms require a user to identify initial parameters. These parameters, or controls, affect overall algorithm performance. Regardless of the anomaly detector being utilized, algorithm performance is often negatively impacted by uncontrollable noise factors which introduce additional variance into the process. In the case of HSI, the noise variables are embedded in the image under consideration. Robust parameter design (RPD) offers a method to model the controls as well as the noise variables and identify robust parameters. This research identifies image noise characteristics necessary to perform RPD on HSI. Additionally, a new data splitting algorithm to predict classifier performance with sparse data sets is presented. Finally, the standard RPD model is extended to consider higher order noise coefficients. Mean and variance RPD models are optimized in a dual response function. Results are presented from simulations as well as applications involving two anomaly detection algorithms, the Reed-Xiaoli anomaly detector and the autonomous global anomaly detector.

Acknowledgments

First and foremost, I would like to thank my family for all of the sacrifices you made for me to accomplish this goal. I am eternally grateful for your constant love and support. It has been a long journey which never would have been possible without you.

Also, I would like to thank my parents for always pushing me to seek excellence and raising me to believe in myself. Knowing that you have always been there for me has been a source of strength.

To my gracious friends, Steve, Stacy and family, thank you for welcoming me into your home to finish up my research. I will be forever indebted to you for your kindness and friendship.

I would also like to express my extreme gratitude and respect to my advisor, Dr. Bauer. Returning to AFIT was a simple choice when I considered working with you again. You have always known when I needed a pick me up or times when I needed to slow down and reconsider situations.

I would like to thank my committee members, especially Maj Mark Friend, whose assistance and encouragement was greatly appreciated as well as all of the other members of the ENS department.

Finally, to the OR crew, Jacob, Earl, Tiffany, Jason, Jeremy, Steve, Ben and Trevor, you have all been there for me whether it was to answer an obscure question (or listen to me talk about the question) or just to add a little humor to an otherwise difficult path.

Frank M. Mindrup

Table of Contents

Page

List of Figures

Figure Page

List of Tables

Table Page

OPTIMIZED HYPERSPECTRAL IMAGERY ANOMALY DETECTION THROUGH ROBUST PARAMETER DESIGN

I. Introduction

1.1 Motivation

The advent of the space age, typically credited to the Soviet Union's launch of Sputnik in October 1957, the emergence of the digital computer and the inception of pattern recognition technology energized a desire to better understand how observations from space could be utilized to perform numerous tasks from weather observations to managing finite Earth resources through imagery. Imagery collected from space could cover large areas. However, the resolution required to provide image quality data from space capable of discerning very minute spatial characteristics would be too expensive and the amount of data overwhelming. Spectral variations across several bandwidths collected through multispectral imaging became an appealing dimensionality reduction method [47]. Hyperspectral imagery (HSI), collected from more than just spaceborne sensors, has since emerged as a valuable tool supporting numerous military and commercial missions ranging from identifying enemy vehicles to detecting oil spills and even cancer.

A hyperspectral image, also called an image cube, consists of k spectral bands of an m by n spatial pixel representation of a sensed area. Each pixel in the spectral dimension represents an intensity of energy reflected back to the sensor. Taken together, these spectral dimensions represent a pixel signature. HSI, by its very nature, can provide a method for identifying at most $(d-1)$ unique spectral signals, where

d is the number of independent bands in an HSI image cube. This is $(d-1)$ rather than d because one band is used to define the background or noise present in an image. Since HSI contains typically hundreds of bands, the number of spectral bands for classification can be large although "high dimensional space is mostly empty" [49, pg. 250]. For instance, spectral bands affected by atmospheric absorption contain little useful information and must be removed; bands that are close to each other are typically correlated and provide little to no additional information.

HSI classification processes can be loosely categorized into three types: anomaly detection, signature matching and change detection [27]. All three classification processes attempt to classify individual image pixels into specific categories using statistical, physical or heuristic methods. An anomaly detector is an HSI classification algorithm which attempts to identify pixels that are different from surrounding pixels, or background, as anomalies. Signature matching compares the spectra for a particular pixel with known spectra for materials contained in a spectral library. Change detection identifies changes within a scene occurring over time. Change detection techniques can be performed with or without knowledge of a spectral library [27]. Anomaly detection algorithms are the easiest classification algorithms to implement as they require no *a priori* signature information and are the focus of this research. It is assumed that images are collected in a rural environment and that true anomalies (man-made objects) are sparse with distinct spectral compositions.

Robust HSI classification algorithms are necessary to counter environmental and other effects. For instance, optimal anomaly detection algorithm parameter settings for a particular background, such as desert, might be completely inappropriate for other backgrounds, such as forest. Landgrebe [48] summarized this concept for future hyperspectral algorithms:

> ...what is needed is an analysis process that is robust in the sense that it would work effectively for data of a wide variety of scenes and conditions, and can be used effectively by users rather than only by producers of the technology. The algorithms do not need to be simple, but they must be simple to apply and robust against user problems [48, pg. 419].

Design of experiments methods such as robust parameter design (RPD) can be applied to reduce the overall variations due to image and sensor noise for a selected set of parameters. While robust parameters can reduce classifier variability within a given region of exploration, oftentimes users of the algorithm will attempt to use the classifier outside of the specified region. In the context of anomaly detection for HSI, the algorithm might encounter an image that is "noisier" than the images used in training [53]. Thus, RPD models for anomaly detection algorithms must not only be robust within the design space but also have good extrapolation properties [79].

1.2 Description of Research

The research presented in this dissertation is comprised of three primary focus areas: defining HSI noise variables for RPD, selecting training and test sets when small sample size is encountered and expanding the standard RPD model to consider higher order noise coefficients. These research areas are applied to anomaly detection algorithms but have uses in signature matching as well as a broader generalization to RPD applications. Figure 1.1 combines all three areas in a single research collection. Shaded boxes represent research areas which are described in more detail in the rest of this Section. Numbers in the upper right-hand corner of shaded boxes correspond to the specific area being addressed.

In Figure 1.1, RPD is broken into processes for training and test. The RPD training process estimates a model, $\hat{y}$, approximating the true classifier response, y. The true response is a function of control variables, x, and training image noise variables, z_{tr}. Optimal control settings, x^*, are identified for a given objective function. The

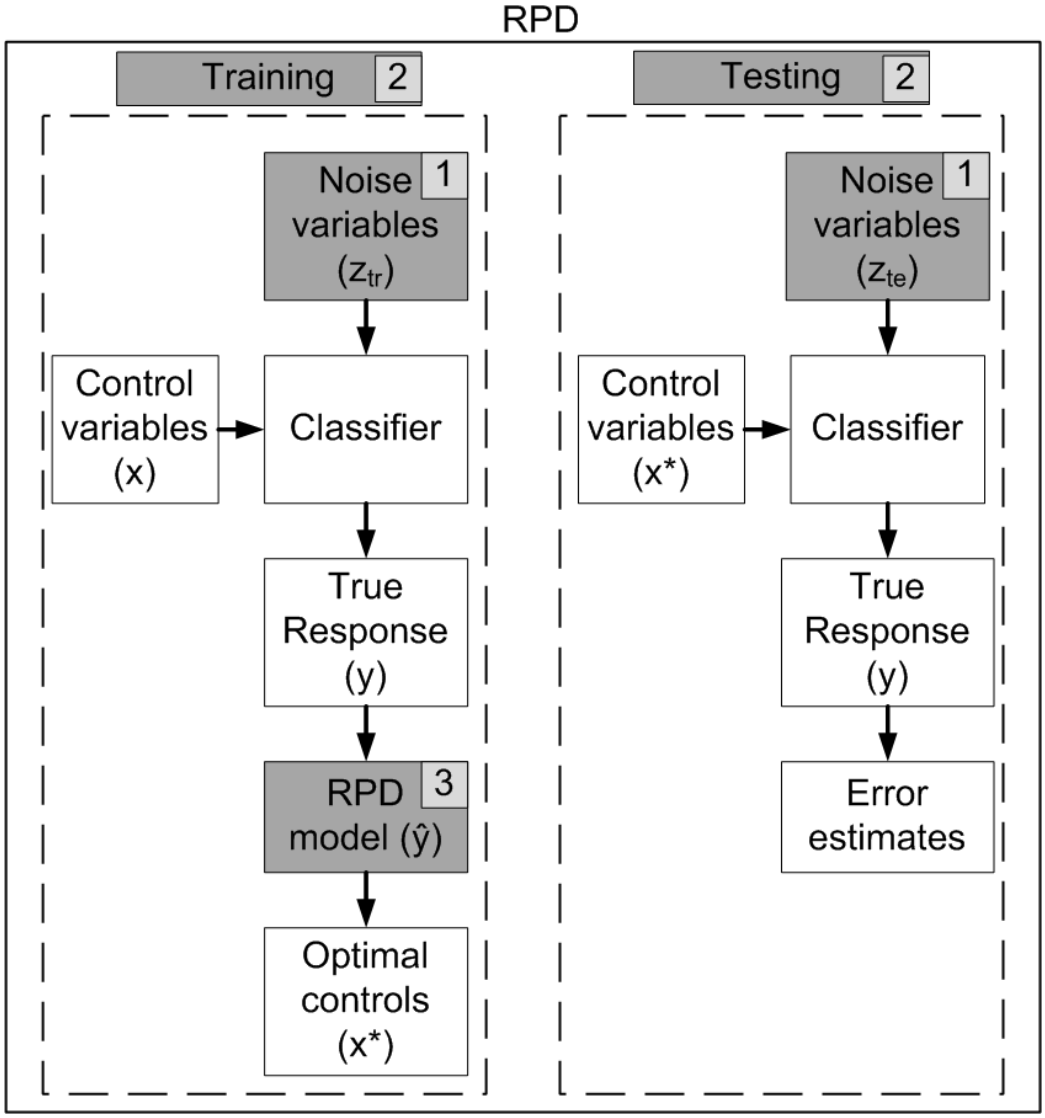


Figure 1.1. Dissertation research focus areas.

test process creates additional responses, y, using the optimal control settings, x^*, and test image noise variables, z_{te}. Measures of error for x^* are used to assess control setting robustness.

The first research area presents a method to uniquely characterize images based on three observable features. Here, certain image features are considered noise variables for RPD models; these characteristics are fixed for a given set of training images, but it is unknown whether all future images will fall within the range of image noise as defined by the training images.

The next area addresses model validation through data splitting. If the set of images available to train the classifier were known to represent all possible noise likely to be observed within an image, training set selection could focus on sets that cover the entire range of potential image noise values. The CADEX algorithm, created by Kennard and Stone [39], selects training sets in this manner. However, since the images (RPD noise variables) used to train the anomaly detection algorithm are not guaranteed to represent every type of hyperspectral image encountered by the classifier, training and test sets of images should be created that "cover approximately the same region and have similar statistical properties" [79, pg. 421]. The DUPLEX algorithm suggested by Kennard [79] creates these similar sets, but the DUPLEX algorithm is intended for problems with large sample sizes. Snee only suggests data splitting when the total number of data points (N) is at least

$$N \geq 2p + 25 \tag{1.1}$$

where p is the "largest number of coefficients one believes will be required to describe the response" [79, pg. 422]. Frequently, the number of images available to train the anomaly detection algorithms falls below this threshold. Many examples exist in the literature with similar small sample size problems [7, 6, 16, 17, 35, 52, 68,

74, 90, 89]. To meet analysis needs, a small sample size training and test selection (SSTATS) method is proposed. This method yields training and test sets that are more representative of one another as assessed by three measures: location, fit and representative error. The SSTATS method can be generalized for use in any problem with small sample size when model validation and prediction are important.

The final research area extends the traditional RPD model. Standard RPD models consider quadratic control terms but assume first order noise terms and control by noise interactions are the only significant noise factors. This assumption was found lacking in an RPD of an anomaly detector. As a result, higher order noise terms are considered and appropriate expected value and variance models are created. These models can be applied to any RPD problem.

1.3 Literature Review of the Topic

The general goal of this research is the identification of robust parameters for anomaly detection algorithms which are capable of consistent performance across a wide variety of images. To this end, this Section presents a broad literature review encompassing overarching concepts germane to this research. Hyperspectral imagery data collection and processing processes are highlighted. Next, anomaly detection algorithm concepts are discussed. Finally, robust parameter design is reviewed including dual response optimization routines. Additional literature review topics are presented in Chapters 2 and 3.

1.3.1 Hyperspectral Imagery.

Hyperspectral sensors utilize information typically collected across contiguous regions of the visible, near-infrared and mid-infrared portions of the electromagnetic spectrum. Hyperspectral remote sensing combines panchromatic imaging and spec-

trometry. Panchromatic imaging focuses on the spatial characteristics of a scene relating to the distribution of the irradiance emitted or reflected over a given spectral band. Spectrometry measures spectral variations of a particular pixel in irradiance. Hyperspectral sensors are capable of collecting both spatial and spectral data simultaneously [27]. Hyperspectral data can then be exploited to remotely identify materials based on their unique spectral compositions [54]. The broad spectrum collected goes beyond the visible spectrum providing more information for classification algorithms to process. For instance, green vegetation has a low reflectance percentage in the visible regions and a much higher reflectance percentage in the infra-red bands of the spectrum where green vegetation actually appears red. Thus, the "health, vigor and canopy cover of green vegetation" [54, pg. 13] can be assessed. In a military context, the infra-red spectral bands can be used to separate green vegetation from camouflage netting. This ability to remotely extract and characterize individual pixels within an image has led to numerous applications including mineral mapping [43], land cover classification [5, 31, 33, 48], urban area classification [8, 70], coastal environment and water quality [11, 21, 61], bathymetry [1], mine detection [92], drug and pollution detection and enforcement [20, 28, 44] and search and rescue applications [27].

Hyperspectral image cubes are generated by collecting the pupil-plane spectral radiance from a spectrometer for each pixel location in an image. A common method of scanning an image to create a 2-dimensional spatial region from airborne or space-based platforms is the push broom imaging approach. In this instance, a spectrometer measures spectral variations for a row of pixels forming a line image at each instance. As the platform moves, new line images are collected and stored until a complete image hypercube is created [27]. The physical dimensions of each individual pixel represent the spatial resolution of the hyperspectral sensor [49].

Complications arise in HSI due to perturbations from environmental and sensor

influences such as weather, time of day, relative humidity, detector response characteristics and imaging angle. These influences greatly impact the reflectance values observed by a sensor requiring sensor calibration and atmospheric compensation techniques to be applied. It is common to apply statistical processing methods to compensate for these issues [27]. Some atmospheric compensation techniques include the empirical line method [77] and the moderate resolution atmospheric transmittance and radiance code (MODTRAN) [9]. Calibration can also be performed using onboard references [91] or other sources [32].

Hyperspectral data requires preprocessing steps before many classification algorithms can be implemented. The most important first step is reducing the dimensionality of the data. Harsanyi and Chang [34] stated most images can actually be described by a small number of dimensions known as the intrinsic dimensionality. This is done by first removing atmospheric absorbtion spectral bands in which most of the energy is absorbed by the atmosphere. Next, principal component analysis (PCA) is often performed to transform the data and reduce the dimensionality into uncorrelated linear combinations of vectors accounting for as much variability in the original data set as possible. The first principal component accounts for the greatest amount of overall variability and subsequent ordered principal components account for successively less variability [24]. A decision must be made to select the number of principal components to retain. Oftentimes, a combined total percentage of variability is selected as a threshold to identify the specified number of components. Another approach considers the number of endmembers or spectrally distinct sources estimated within an image. Chang [15] states that estimating the true number of spectrally distinct signal sources in an HSI image is difficult. Particularly, when well structured high-dimensional data are encountered, the data tend to be distributed in a much lower dimensional space.

Independent component analysis (ICA) is another transformation commonly applied to HSI data. As the name implies, ICA is intended to recover independent sources from unknown linear mixtures of unobserved independent sources or spectra [87]. It is assumed that the true spectra are statistically independent with non-Gaussian distributions and combined through a linear mixture when collected by a hyperspectral sensor. Further, it is assumed that there are at least as many spectral bands as true endmembers within an image. Assume there are n linear mixtures, $x = (x_1, x_2, \ldots, x_n)^T$, of n independent components. Also, consider n random vectors, $s = (s_1, s_2, \ldots, s_n)^T$, representing "latent variables" [36] meaning the random vectors cannot be directly observed. In matrix form, the problem becomes

$$x = As \tag{1.2}$$

where A is an assumed unknown $n \times n$ linear mixture matrix. Estimating for A and inverting yields W which can then be used to solve for the latent variables in the following manner [36]:

$$s = Wx. \tag{1.3}$$

Methods of solving for the inverse of the mixing matrix, W, can involve complex computations such as solving for the negentropy, measuring a random variable's entropy in comparison with a Gaussian variable. A negentropy of zero indicates the random variable is distributed approximately Gaussian while positive values indicate non-Gaussian distributions [36].

1.3.2 Anomaly Detection.

Anomaly detectors, also known as outlier detectors or novelty detectors, are HSI classifiers used to detect objects that are statistically or geometrically different from

the image background [27]. Anomalies are identified based on a background model, either local or global. Local background models compare each pixel with neighboring pixels providing the ability to identify isolated targets in the open. This characteristic makes local background models susceptible to false alarms if true anomalies encompass a vast majority of the neighborhood used to describe the neighborhood background. Global background models compare pixels with an estimate of the background of the entire image or a large area of the image. Global models minimize the false alarm rate observed in local background models. However, the global background models can have a difficult time identifying isolated targets [81].

A common assumption for anomaly detectors, whether the local or global background model is used, is that the hyperspectral data follow a Gaussian distribution. The generalized likelihood ratio test is often applied to test for the existence of anomalies within an image [81]. Some local background models are the Reed-Xiaoli (RX) [67] detector (described further in Chapter II), the locally adaptive iterative RX detector [84], the support vector data description (SVDD) [7] as well as numerous other variations of the RX detector. Some examples of global background models include the Gaussian mixture model generalized likelihood ratio test (GMM-GLRT) [80], orthogonal subspace projection RX [13] and the autonomous global anomaly detector (AutoGAD) [37] (described further in Chapter III). Additional anomaly detection algorithms based on concepts such as Bayesian classifiers, clustering, kernels and other methods are found in the literature [4, 12, 26, 30, 76].

At the most basic level, anomaly detection applications can be considered a "binary hypothesis testing problem" [54]. Expanding mathematically on the anomaly detection concept, consider a generic anomaly detection system, A, with a forced decision mapping an event, $e \in \mathcal{E}$, first to a feature vector, $f \in \mathcal{F}$, then to a label, $l \in \mathcal{L}$. There are two possible mutually exclusive events, $\mathcal{F}_{\text{present}}$ or $\mathcal{F}_{\text{not}}$ resulting in two pos-

sible mutually exclusive labels, $\mathcal{L} = \{p, np\}$ where p and np denote present and not present respectively. Consider features $f^{(1)}, f^{(2)} \in \mathcal{F}$ such that $f^{(1)} \neq f^{(2)}$ where $f^{(1)}$ maps to label p and $f^{(2)}$ maps to label np. Next, consider a new event, $e \in \mathcal{E}$, yielding a feature vector $f \in \mathcal{F}$ for which a label is to be assigned using the anomaly detection system, A, and a metric $d \in \mathcal{D}$ denoting any metric defined on the set of features, $\mathcal{F}$ with $\mathcal{D}$ representing all possible metrics where $\mathcal{D} = \{d : \mathcal{F}^2 \to \mathbb{R} | d \text{ is a metric}\}$. The truth mapping from any feature vector, $f \in \mathcal{F}$, to a label, $l \in \mathcal{L}$, is defined as T. This process shown in Figure 1.2 represents an anomaly detection problem. The

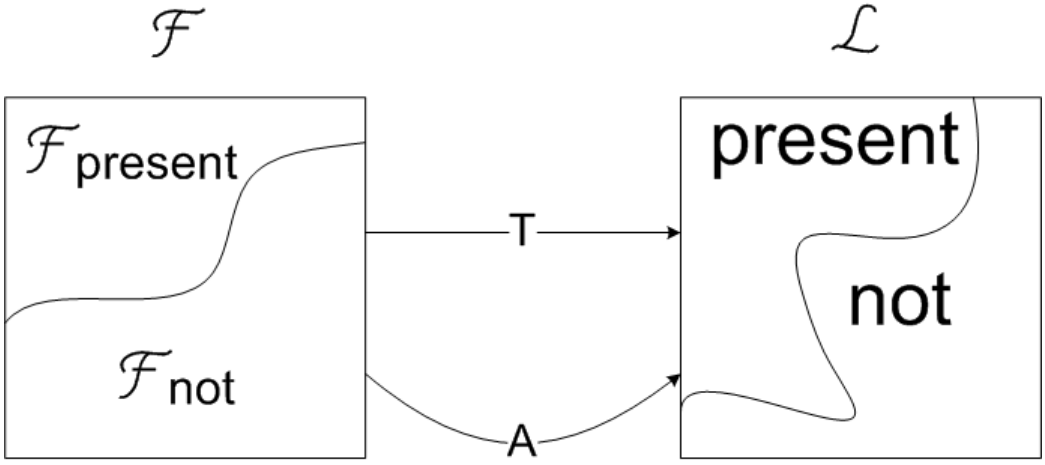


Figure 1.2. Target detection problem.

anomaly detection system, $A : \mathcal{F} \to \mathcal{L}$, is then defined as [86]:

$$A(f) = \begin{cases} p & \text{if} \quad d(f, f^{(1)}) < d(f, f^{(2)}) \\ np & \text{if} \quad d(f, f^{(1)}) \geq d(f, f^{(2)}) \end{cases} \tag{1.4}$$

1.3.3 Robust Parameter Design.

Genichi Taguchi proposed an innovative parameter design approach for reducing variation in products and processes in the 1980's. His methods were quickly adopted across several industries but eventually met with contention over several issues such as

confounding and experimental design size, to name a few [63]. As a result, a response surface approach was also developed. Taguchi's methods are still applied and thus both methods will be described in more detail in the sections that follow. Taguchi's methods are especially useful when the true model is expected to be first order in both control and noise variables with control by noise interactions [59].

Montgomery [59] describes RPD as an approach to experimental design that focuses on selecting control factor settings that optimize a selected response while minimizing the variance due to noise factors. Control factors are those factors that can be modified in practice while noise factors are unexplained or uncontrollable in practice. These noise factors can typically be controlled during research and development allowing RPD to be performed. Some cited examples of noise factors are environmental factors such as temperature or relative humidity, properties of raw materials and process variables difficult to control or maintain at a specified target. Montgomery [59] further identified four focuses for RPD:

1. Design systems that are insensitive to environmental factors that can affect performance once the system is deployed.

2. Design products insensitive to variability due to system components.

3. Design processes so the manufactured product is as close as possible to desired target specifications.

4. Determine operating conditions for process so critical process characteristics are close to desired target values and variability around this target is minimized.

An RPD problem only exists if there is at least one interaction between a control and noise factor. If a control by noise factor interaction does not exist, the variance will be constant across the entire range of control variables. In this situation, classical

approaches to optimizing a process response can be applied without regard to noise. If a control by noise interaction does exist, then there is a control setting that will minimize the variance across the range of noise variables. When a control by noise interaction exists creating an RPD problem, control factors can be classified into three categories: location factors where control factors effect the process mean as they are varied across their range, dispersion factors if a control factor effects the process variance and a combination where a control factor impacts both the mean and variance of the process [63].

1.3.4 Taguchi's Method - Crossed Array Designs.

Taguchi's method is centered on orthogonal designs. Montgomery [59] defines an orthogonal design as one in which the columns of the design matrix, X, are orthogonal meaning that their inner product sums to zero. Orthogonal designs are useful in designed experiments because they allow the experimenter to examine individual effects of each factor in the design matrix. As the number of experimental runs increases in the X matrix, the potential number of factors, interactions and higher order effects available to be estimated also increases. Taguchi's crossed array used orthogonal arrays of the control variables, called the inner array, and crossed them with orthogonal arrays of the noise factors, known as the outer array.

Taguchi summarized the output from his design using two summary statistics, the mean response and signal-to-noise ratio (SNR). Taguchi's SNR was defined based on the goal of the experiment. If the experimenter wanted to minimize the response, smaller is better (SNR_s), then the following SNR should be utilized

$$SNR_s = -10 \log \sum_{i=1}^{n} \frac{y_i^2}{n} \tag{1.5}$$

where n is the number of outer array replications of the response, y_i, to be summed.

If the experimenter wished to maximize the response meaning a larger response is better (SNR_ℓ), the SNR was changed by calculating the squared reciprocal of the response as shown in the following formula.

$$SNR_\ell = -10 \log \sum_{i=1}^{n} \frac{1/y_i^2}{n} \tag{1.6}$$

If there is a specific target value desired (SNR_{T_1}), the following formula can be applied

$$SNR_{T_1} = -10 \log s^2 \tag{1.7}$$

where s^2 is the variance of the outer array replications of the response, y_i, from the target defined as $s^2 = \sum_{i=1}^{n} (y_i - \bar{y})^2 / (n-1)$ [63]. This target SNR can be further defined (SNR_{T_2}) in cases where the response standard deviation is related to the mean as

$$SNR_{T_2} = -10 \log \left(\frac{\bar{y}^2}{s^2} \right) \tag{1.8}$$

In all SNR cases, the SNR value is maximized. Thus, analysis consists of calculating the mean response and SNR for factors at different settings and identifying which settings optimize the response while minimizing variance. SNR_{T_2} is the only true SNR as it is dimensionless.

Taguchi's arrays only consider main effects and first-order interactions. If there are higher order terms required in the model, Taguchi's method will misspecify the model. Finally, none of the SNRs are able to separate effects strictly due to the mean or the variance as multiple control factor settings could produce the same SNR. Therefore, it is often considered more appropriate to model the variance and mean model separately as is shown in the next Section [63].

1.3.5 Response Surface Model Method - Combined Array Designs.

The combined array or response surface model (RSM) approach applies more emphasis to learning the characteristics of the true process rather than the optimization of a criterion. RSM methods focus on the roles of control variables on mean and variance in order to provide an estimate of the mean and variance at any location of interest defined in the control variables. Typically, second-order models are developed when using RSM approaches and higher order interactions and terms are ignored due to the sparcity of effects principle. Further, noise terms, z, are assumed to be independent ($\text{cov}(z_i, z_j) = 0 \quad \forall \quad i \neq j$) implying no noise by noise interaction terms are significant. A general matrix form of the quadratic response surface model is in the following Equation [22]:

$$y = G(x, z) = \beta_0 + x'\beta + x'Bx + z'\gamma + x'\Delta z + \epsilon \tag{1.9}$$

where x is an $r_{\mathbf{x}} \times 1$ vector of control variables, z is an $r_{\mathbf{z}} \times 1$ vector of noise variables, β_0 is the intercept, β is an $r_{\mathbf{x}} \times 1$ vector of control variable coefficients, B is an $r_{\mathbf{x}} \times r_{\mathbf{x}}$ matrix of the quadratic control coefficients, γ is an $r_{\mathbf{z}} \times 1$ vector of noise variable coefficients, Δ is an $r_{\mathbf{x}} \times r_{\mathbf{z}}$ matrix of control by noise interaction coefficients and ϵ is a random error assumed to be normally distributed, $N(0, \sigma^2 I_{r_{\mathbf{z}}})$; $r_{\mathbf{x}}$ and $r_{\mathbf{z}}$ represent the number of control and noise factors respectively. The noise variables, $z = (z_1, z_2, \ldots, z_{r_z})$, are assumed to be a vector of independent random variables with $E(z_i) = 0 \quad \forall \quad i$ and $var(z) = \sigma_z^2 I_{r_z}$ which is easily accomplished by centering and scaling. Thus the general form of the mean model only includes the control variables and is shown in Montgomery [59] to be

$$E(y|x) = E_{z,\epsilon}\left(G(x, \cdot)|x\right) = \beta_0 + x'\beta + x'Bx. \tag{1.10}$$

where $E(y|x)$ is short-hand notation for $E_{z,\epsilon}\left(G(x,\cdot)|x\right)$ which will be used throughout the remainder of this Section. Likewise, the variance model can be found by treating z as a random variable and applying the variance operator to Equation (1.9). The variance model becomes

$$var(y|x) = var_{z,\epsilon}\left(G(x,\cdot)|x\right) = \sigma_z^2\left(\gamma + \Delta' x\right)'\left(\gamma + \Delta' x\right) + \sigma^2 \tag{1.11}$$

where σ^2 is the variance of ϵ, typically estimated as the Mean Square Error found from performing a regression on the design, σ_z^2 is the variance-covariance matrix of z typically assumed to be the identity matrix and $var(y|x)$ represents $var_{z,\epsilon}\left(G(x,\cdot)|x\right)$ and will be used throughout the remainder of this Section [63].

1.3.6 Dual Response Surface Optimization.

Often, robust control settings are chosen by solving an optimization problem that achieves a target mean while minimizing the variance. One optimization approach suggested by Myers and Montgomery [63] is

$$\begin{aligned} \min_{x\in\mathbf{D}} \quad & var\left(y|x\right) \\ s.t. \quad & E\left(y|x\right) = T \end{aligned} \tag{1.12}$$

where T is a target value for the mean and the control parameters are confined to the experimental design region, $\mathbf{D}$, which is a closed and bounded compact set. Before continuing, let the mean model be estimated by

$$\hat{\mu}_y = \hat{E}\left(y|x\right) \tag{1.13}$$

and the variance model by

$$\hat{\sigma}_y = \widehat{var}\,(y|x)\,. \tag{1.14}$$

Myers and Montgomery [63] suggest the use of overlays of contour plots for the mean and variance surfaces to select optimal control settings. This method has merits by allowing a visual assessment of the tradeoffs between the mean and variance for a given algorithm but is limited to two control variables.

Myers and Carter [62] and Vining and Myers [88] applied Lagrangian multipliers in an attempt to combine the mean and variance models into a single objective function. The authors included an additional constraint limiting the optimal control factors to a spherical region with $x'x = \rho^2$ where ρ is the radius of the spherical region. The Lagrangian function is described as

$$L = \hat{\sigma}_y - \lambda_\theta(\hat{\mu}_y - T) - \lambda_p(x'x - \rho^2) \tag{1.15}$$

where λ_θ is the weighting applied to the difference between the mean and its target value and λ_p is the weighting applied to the spherical region constraint [53].

Del Castillo and Montgomery [23] implemented the generalized reduced gradient (GRG) algorithm to solve the Lagrangian function in Equation 1.15. The GRG allowed inequality constraints yielding local optima. The equality constraints implemented in Equation (1.15) were not always guaranteed to produce local optima.

Lin and Tu [51] developed a method to identify robust settings using the response surface methodology. This method simultaneously reduced the variance while improving the mean response value by considering a target mean. Three different measures for mean squared error (MSE) were suggested depending on the response value; in all cases, the MSE is minimized. When a smaller response is desired (MSE_s), the Lin

and Tu criterion becomes

$$MSE_s = \hat{\mu}_y^2 + \hat{\sigma}_y^2. \tag{1.16}$$

Similarly, when a larger response of interest is desired (MSE_ℓ), the criterion is

$$MSE_\ell = -(\hat{\mu}_y^2) + \hat{\sigma}_y^2. \tag{1.17}$$

Finally, when a desired target mean, T, is specified (MSE_T), the criterion becomes

$$MSE_T = (\hat{\mu}_y - T)^2 + \hat{\sigma}_y^2. \tag{1.18}$$

Shaibu and Cho [73] extended the Lin and Tu MSE approach to include a target standard deviation in the equations. As in the Lin and Tu method, three methods are proposed based on the desired response value. The authors included a constraint for an upper bound on variance, S. If a smaller response is desired (MSE_s), the Shaibu and Cho proposed criterion is

$$MSE_s = \hat{\mu}_y + (\hat{\sigma}_y - T_S)^2 \tag{1.19}$$

where T_S is the user-specified target standard deviation. When a larger response is desired (MSE_ℓ), the criterion becomes

$$MSE_\ell = -\left[\hat{\mu}_y + (\hat{\sigma}_y - T_S)^2\right]. \tag{1.20}$$

Finally, the Shaibu and Cho criterion when a target mean, T, is specified (MSE_T) becomes

$$MSE_T = (\hat{\mu}_y - T)^2 + (\hat{\sigma}_y - T_S)^2. \tag{1.21}$$

Copeland and Nelson [19] restricted the distance between the observed mean re-

sponse value and the target value. When a target mean value is desired, the authors suggest minimizing an objective function specified as $\hat{\sigma}_y + \varepsilon$ by

$$\varepsilon = \begin{cases} (\hat{\mu}_y - T)^2 & \text{if } (\hat{\mu}_y - T)^2 > \Delta^2 \\ 0 & \text{if } (\hat{\mu}_y - T)^2 \leq \Delta^2 \end{cases} \tag{1.22}$$

where Δ^2 is a user-specified bound on the difference between the observed mean and the mean target value.

Tang and Xu[85] applied goal programming to the dual response problem. The Tang and Xu dual response problem is

$$\begin{aligned} \min_{x} \quad & \delta_\mu^2 + \delta_\sigma^2 \\ s.t. \quad & \hat{\mu}_y - w_\mu \delta_\mu = T \\ & \hat{\sigma}_y - w_\sigma \delta_\sigma = T_S \\ & x'x \leq \rho \text{ or } x_l \leq x \leq x_u \end{aligned} \tag{1.23}$$

where the δ terms in the objective function are unrestricted scalar variables representing slackness and the w terms are user-specified weights [53].

Several other applications for solving the dual response surface optimization problem have been proposed. Kim and Lin [40] presented fuzzy optimization methods. Pareto optimal solutions were discussed by Koksoy and Dogamaksoy [42] and Lam and Tang [45]. Table 1.1 summarizes most of the dual surface optimization methods described in this literature review. The research presented in this dissertation focused on the Lin and Tu approach to dual response optimization although other methods were considered.

Table 1.1. Methods for solving RPD dual response problem.

Reference	**Target is best**	**Smaller the better**	**Larger the better**
Myers & Montgomery (2002)	$\min_x \hat{\sigma}_y$ $s.t. \quad \hat{\mu}_y = T$		
Vining & Myers (1990)	$\min_x \hat{\sigma}_y$ $s.t. \quad \hat{\mu}_y = T$ $x'x = \rho$	$\min_x \hat{\mu}_y$ $s.t. \quad \hat{\sigma}_y = T_S$ $x'x = \rho$	$\max_x \hat{\mu}_y$ $s.t. \quad \hat{\sigma}_y = T_S$ $x'x = \rho$
Del Castillo & Montgomery (1993)	$\min_x \hat{\sigma}_y$ $s.t. \quad \hat{\mu}_y = T$ $x'x \leq \rho$	$\min_x \hat{\mu}_y$ $s.t. \quad \hat{\sigma}_y = T_S$ $x'x \leq \rho$	$\max_x \hat{\mu}_y$ $s.t. \quad \hat{\sigma}_y = T_S$ $x'x \leq \rho$
Lin & Tu (1995)	$\min_x (\hat{\mu}_y - T)^2 + \hat{\sigma}_y^2$	$\min_x \hat{\mu}_y^2 + \hat{\sigma}_y^2$	$\min_x -\hat{\mu}_y^2 + \hat{\sigma}_y^2$
Copeland & Nelson (1996)	$\min_x (\hat{\mu}_y - T)^2 + \hat{\sigma}_y^2$ s.t. $(\hat{\mu}_y - T)^2 \leq \Delta^2$	$\min_x \hat{\mu}_y^2 + \hat{\sigma}_y^2$ s.t. $(\hat{\mu}_y - T)^2 \leq \Delta^2$	$\min_x -\hat{\mu}_y^2 + \hat{\sigma}_y^2$ s.t. $(\hat{\mu}_y - T)^2 \leq \Delta^2$
Shaibu & Cho (2009)	$\min_x (\hat{\mu}_y - T)^2 + (\hat{\sigma}_y - T_S)^2$ s.t. $\hat{\sigma}_y \leq S$	$\min_x \hat{\mu} + (\hat{\sigma}_y - T_S)^2$ s.t. $\hat{\sigma}_y \leq S$	$\min_x -\left[\hat{\mu}_y + (\hat{\sigma}_y - T_S)^2\right]$ s.t. $\hat{\sigma}_y \leq S$
Tang & Xu (2002)	$\min_x \quad \delta_\mu^2 + \delta_\sigma^2$ $s.t. \quad \hat{\mu}_y - w_\mu \delta_\mu = T$ $\hat{\sigma}_y^2 - w_\sigma \delta_\sigma = T_S$ $x'x \leq \rho$ or $x_l \leq x \leq x_u$		

1.4 Original Contributions and Research Overview

This research makes original contributions in both statistics and HSI. Relative to HSI, specific image noise characteristics are defined to uniquely describe hyperspectral images. A small sample data splitting algorithm is developed to create representative training and test sets essential for algorithm performance estimation. In statistics, the RPD model is extended to include higher order noise terms. Table 1.2 maps the chapters of this dissertation to the particular RPD area of study.

Table 1.2. Chapter description.

Chapter	Image characteristics	Data splitting	RPD extensions
2	x	x	
3			x

Chapter 2 presents a novel data splitting method utilizing discrete and continuous image characteristics as representations of the noise present in HSI. Specifically, the number of clusters, Fisher ratio and percent of target pixels are used to characterize HSI. The chapter also develops the small sample training and test set selection (SSTATS) method to identify training and test sets for use in RPD of HSI. The training and test sets provide excellent separation of observed noise characteristics. The SSTATS method is compared with the CADEX and DUPLEX algorithms proposed by Kennard [39, 79] as well as random selection approaches to data splitting. Results from simulations as well as an application using the RX algorithm display the superiority of the SSTATS algorithm.

Chapter 3 expands the traditional RPD model to include noise by noise interactions and squared noise terms. These coefficients are typically assumed to be negligible, but were significant in an RPD of the anomaly detection algorithm, AutoGAD. The RPD mean and variance models are extended to include the higher order noise terms. The Lin and Tu MSE approach [51] to solving the RPD problem is utilized to

select robust control settings. The mean and variance models including higher order noise coefficients can be applied to any dual response surface optimization technique listed in Table 1.1.

II. Small Sample Training and Test Selection Method for Optimized Anomaly Detection Algorithms in Hyperspectral Imagery

2.1 Introduction

There are typically two broad classes of unsupervised anomaly detectors considered in the literature depending on the background estimate. Local models define the background based on a local neighborhood around a test pixel while global models typically specify a background distribution from across the entire image, or a large section of the image [81]. A brief list of anomaly detection algorithms proposed for hyperspectral imagery (HSI) include the support vector data description algorithm (SVDD) [7], the Reed-Xiaoli (RX) [67] algorithm, the locally adaptive iterative RX detector [84] and the autonomous global anomaly detector (AutoGAD) [37]. Most, if not all, anomaly detection algorithms require a user to identify some initial parameters. These parameters (or controls) affect the overall algorithm performance.

Anomaly detectors are relatively simple to implement as they require no *a priori* signature information. These algorithms are intended for images with sparse anomalies. Regardless of the anomaly detector being utilized, algorithm performance is often negatively impacted by uncontrollable noise factors which introduce additional variance into the process. A generic anomaly detector is depicted in Figure 2.1. A vector of control variables are input into the anomaly detector (classifier) producing a response, y. A vector of uncontrollable noise variables also affect the classifier output. The noise variables are considered uncontrollable in real-world applications, but can be fixed for a designed experiment. In the case of HSI, the noise variables are embedded in the image under consideration. For instance, two images of the same scene taken at different times of day will have different sun angle effects introducing

variability into the spectral data collected [47]. Thus, the need arises to identify robust anomaly detector settings capable of yielding consistent responses across varied image backgrounds. Landgrebe [48] summarized this concept for future hyperspectral algorithms:

> ...what is needed is an analysis process that is robust in the sense that it would work effectively for data of a wide variety of scenes and conditions, and can be used effectively by users rather than only by producers of the technology. The algorithms do not need to be simple, but they must be simple to apply and robust against user problems [48, pg. 419].

Consider the performance of an anomaly detector, y, represented as a function of control and noise variables with some random process noise in Equation (2.1). Figure 2.1 depicts this process. As a result, HSI anomaly detection fits directly into the robust parameter design (RPD) framework.

$$y = F(x, z) + \epsilon \tag{2.1}$$

Taguchi developed RPD as a means to identify optimal algorithm parameters by targeting a specified process mean while minimizing process variance. Taguchi's RPD method utilizes orthogonal designs by crossing an orthogonal array of control variables with an orthogonal array of noise variables. Some authors have voiced concern with aspects of Taguchi's work; one proposed correction led to a combined array approach in Myers [63] which will be described here [64, 69].

Training and test sets of hyperspectral images are typically selected randomly to assess algorithm performance. Davis [22] considered each training image as a categorical noise variable in his RPD for HSI anomaly detection. This requires RPD selected optimal settings based on each training image. For anomaly detection applications, it would take considerable planning to adjust anomaly detector control settings based

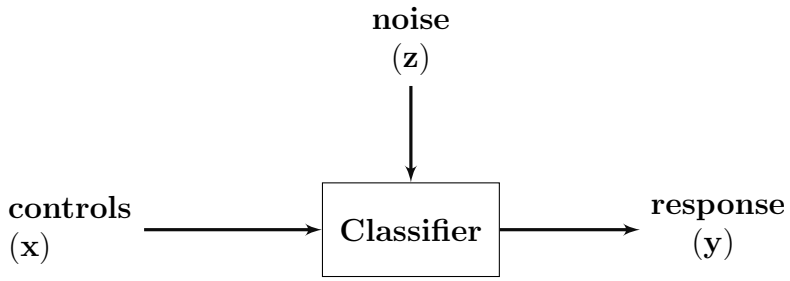


Figure 2.1. Nominal anomaly detector.

on each incoming image to be processed. Mindrup *et al.* [57] developed a framework of continuous and discrete noise characteristics to describe images based on three measurable noise characteristics: the Fisher score, the ratio of target pixels and the number of clusters. These are not the only characteristics observable within an image, but rather a subset that are easily calculated within a training set with truth information. Thus, a crossed array of control variables with observed noise characteristics was possible. This crossed design array was used in RPD to identify robust control settings. Unfortunately, the selected image noise features were a result of observational data and are considered "messy" [39] as the images do not typically separate into an orthogonal training and test set. Mindrup *et al.* [57] proposed a greedy heuristic to select a training set covering the largest range in each noise variable. The heuristic yielded multiple optimal training sets in most cases. Kennard and Stone [39] developed the CADEX algorithm to assist developing experimental designs for response surface exploration. This algorithm primarily focuses on performing "reasonable smoothing of the results and to have plans as model-free as possible" [39]. The algorithm focuses on developing a robust training set which often yields training and test sets that are not representative of one another. Kennard improved upon his initial approach with the DUPLEX algorithm which "provides a more stringent method of model validation because some extreme points appear in both the estimation and prediction sets" [79]. Snee suggests data splitting only when the total

number of points available for the training and test set (N) is greater than twice the number of parameters being estimated (p), or more specifically, $N \geq 2p + 25$. In the case of HSI, sometimes the number of images available for training and testing algorithms falls well below this benchmark. This section proposes a training and test selection strategy intended for small data sets where $N \leq 30$. While splitting small sample sizes yields estimated coefficients with larger variance than those obtained by fitting the entire data set [79], data splitting is necessitated by the nature of the HSI problem under consideration. In general, this chapter seeks to find training and test sets that are as similar as possible in order to avoid bias when assessing the different algorithms.

The work in this chapter extends the work found in Mindrup *et al.* [57] and Mindrup *et al.* [58]. The chapter develops the small sample training and test selection (SSTATS) method for selecting unique optimal training and test subsets of hyperspectral images yielding consistent RPD results across both subsets. SSTATS is based on measures such as the D-optimal score and distance norms. These subsets are not necessarily orthogonal since they are formed using observational data, but still provide improvements over random training and test subset assignments by maximizing the volume and average distance between image characteristics. Further, the SSTATS training and test sets are more "representative" of one another when compared with subsets generated using the CADEX and DUPLEX algorithms on datasets with small sample sizes. Representative training and test sets are necessary as models are often used on data collected outside of the bounds specified by the training set.

The remainder of this chapter is organized as follows. First, robust parameter design concepts are reviewed. Then the CADEX and DUPLEX training and test selection methods and previously published HSI noise variable creation methods are reviewed. Next, the SSTATS training and test selection strategy is developed. A

simulation experiment for non-orthogonal noise variables reveals the utility of optimal training and test sets as compared with randomly selected training and test sets. Following the simulation, all of the training and test selection methods are compared in a real-world example by using RPD and the selected training and testing sets to select robust control parameters for the RX anomaly detection algorithm.

2.2 RPD Background

Regression models are typically vague with respect to what transformations are required of the factors, the existence of asymptotes and the fact that most experiments contain multiple responses [39]. RPD methods attempt to identify robust process control settings capable of consistent performance by incorporating the mean and variance into a single response variable, even in the face of uncontrollable or noise factors. It is assumed that noise factors are uncontrollable in practice, but can be controlled for designed RPD experiments [63]. Further, it is assumed that the overall true process response, y, can be described as a function of control variables, x, and noise variables, z

$$y = G(x, z). \tag{2.2}$$

Lin and Tu [51] proposed a criterion considering the process mean and variance as an estimate for mean square error (MSE) to solve for optimal control variable settings in RPD problems. The Lin-Tu (LT) MSE minimization criterion considers the process mean based on a target value, T, and process variance both conditioned with respect to x, as shown below.

$$LT_{z,\epsilon}\left(G(x,\cdot)|x\right) = \left\{E_{z,\epsilon}\left(G(x,\cdot)|x\right) - T\right\}^2 + var_{z,\epsilon}\left(G(x,\cdot)|x\right) \tag{2.3}$$

The vector of optimal control variable settings, x^*, can be identified by solving

the following constrained optimization problem

$$x^* = \arg\min_{x \in \mathbf{D}} \quad LT_{z,\epsilon}\left(G(x,\cdot)|x\right) \tag{2.4}$$

where the vector of control variables, x, is constrained to the experimental design space, $\mathbf{D}$, which is a closed and bounded compact set.

Typically, second-order models are developed in response surface methodology approaches to RPD and higher order control interactions are ignored due to the sparsity of effects principle [63]. Noise by noise interactions and squared noise terms are also assumed to be negligible. A general matrix form of the quadratic response surface model proposed by Myers [63] is

$$y = G(x,z) = \beta_0 + x'\beta + x'Bx + z'\gamma + x'\Delta z + \epsilon \tag{2.5}$$

where x is an $r_{\mathbf{x}} \times 1$ vector of control variables, z is an $r_{\mathbf{z}} \times 1$ vector of noise variables, β_0 is the intercept, β is an $r_{\mathbf{x}} \times 1$ vector of control variable coefficients, B is an $r_{\mathbf{x}} \times r_{\mathbf{x}}$ matrix of the quadratic control coefficients, γ is an $r_{\mathbf{z}} \times 1$ vector of noise variable coefficients, Δ is an $r_{\mathbf{x}} \times r_{\mathbf{z}}$ matrix of control by noise interaction coefficients and ϵ is a random error assumed to be normally distributed $N(0, \sigma^2 I_{r_{\mathbf{z}}})$; $r_{\mathbf{x}}$ and $r_{\mathbf{z}}$ represent the number of control and noise factors respectively. The noise variables, $z = (z_1, z_2, \ldots, z_{r_z})$, are assumed to be a vector of independent random variables with $E(z_i) = 0 \quad \forall \quad i$ and $var(z) = \sigma_z^2 I_{r_z}$ which is easily accomplished by centering and scaling. The mean model with respect to z for the estimated quadratic model in Equation (2.5) becomes

$$E(y|x) = E_{z,\epsilon}\left(G(x,\cdot)|x\right) = \beta_0 + x'\beta + x'Bx. \tag{2.6}$$

where $E(y|x)$ is short-hand notation for $E_{z,\epsilon}\left(G(x,\cdot)|x\right)$ and will be used throughout the remainder of this Chapter.

Similarly, the variance model of Equation (2.5) is given by

$$\begin{aligned} var(y|x) &= var_{z,\epsilon}\left(G(x,\cdot)|x\right) \\ &= \left(\gamma' + x'\Delta\right) var_z(z) \left(\gamma' + x'\Delta\right)' + \sigma^2 \\ &= \sigma_z^2 \left(\gamma' + x'\Delta\right)\left(\gamma' + x'\Delta\right)' + \sigma^2. \end{aligned} \tag{2.7}$$

The corresponding LT criterion becomes

$$\begin{aligned} LT(y|x) &= LT_{z,\epsilon}\left(G(x,\cdot)|x\right) \\ &= \left(\beta_0 + x'\beta + x'Bx - T\right)^2 + {\sigma_z}^2 \left(\gamma' + x'\Delta\right)\left(\gamma' + x'\Delta\right)' + \sigma^2. \end{aligned} \tag{2.8}$$

For the remainder of this section, the notation $LT(y|x)$ will be used to represent $LT_{z,\epsilon}\left(G(x,\cdot)|x\right)$.

The noise parameters, z, effect the overall LT criterion in the variance model through the noise parameter coefficients, γ and Δ, but the criterion is completely in terms of control parameters, x. Thus, optimal control settings can be identified through constrained optimization as in Equation (2.4).

2.3 Training and Test Set Selection

The general training set selection problem is exemplified by considering the images represented by noise variables in Figure 2.2. Common practice assumes orthogonal noise features such as a replicate of the 2^2 factorial design represented by circles in Figure 2.2. Typically, this is possible in industrial applications by identifying high and low noise settings that can be fixed in a test environment. HSI noise characteristics

cannot be fixed in the same manner. When considering a finite number of images, the noise within a set of hyperspectral images tends to look more like the observed points depicted as triangles in Figure 2.2. It is not readily apparent how to select a representative training and test set from the observed points. Randomly separating the data would not necessarily guarantee that "some of the points in the [training] set are extrapolation points, and the [test] set would provide no information on how well the model is likely to extrapolate" [60, pg. 311]. In general, the training set is used to parameterize an RPD model and the test set provides an opportunity to assess how well the model predicts performance. In what follows, three procedures are presented as candidates for selecting training and test sets.

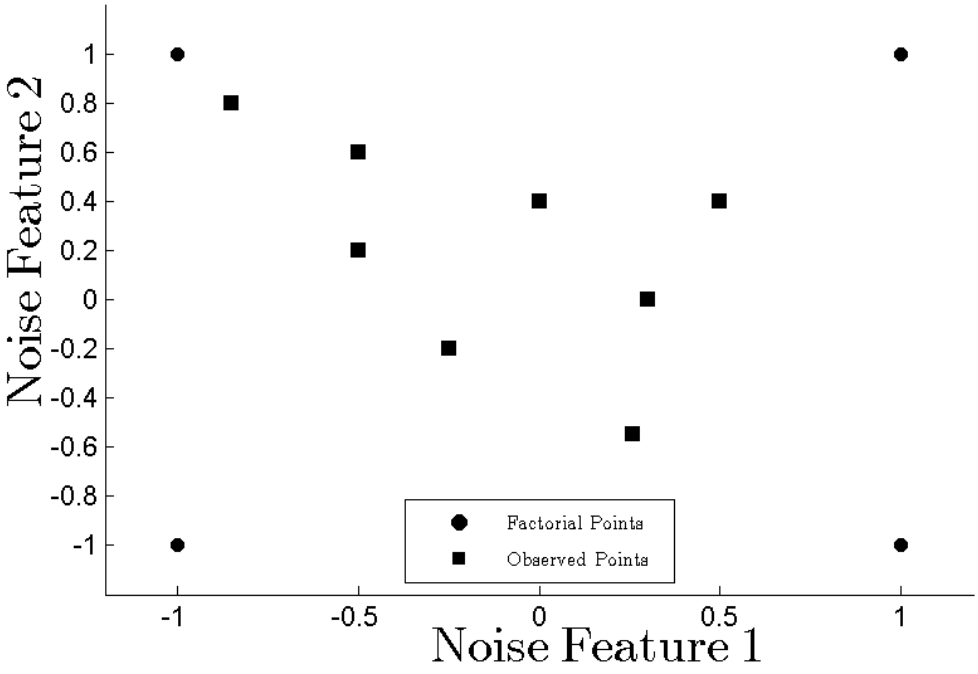


Figure 2.2. Generic training set selection problem.

Identifying a robust training and test set of images can be considered a combinatorial optimization problem. Formally, a combinatorial optimization problem aims to select an object from a finite or countably infinite set. In terms of selecting training

sets of hyperspectral images, the combinatorial optimization problem is comprised of a pair (Ω, f), where Ω consists of all possible combinations of images and f is the cost function used to select the optimal combination [65]. Below, various strategies of searching Ω are presented.

In the remainder of this section, the CADEX and DUPLEX algorithms are reviewed. Then HSI noise characteristics are defined and some notation is presented providing an avenue to separate images into training and test sets. Finally, the proposed small sample training and test selection (SSTATS) method is developed.

2.3.1 CADEX.

In what follows, the CADEX algorithm is described following the description by Kennard and Stone [39]. First, let p represent the number of control factors $(x_1, x_2, \ldots, x_p)$. Further, there are $n \leq N$ distinct points to be chosen for the experimental design from the total possible candidate design points, N, contained in the $p-$dimensional design space spanned by the factors. The N candidate design points can be represented in a matrix X as

$$X = \begin{bmatrix} x_{11} & x_{21} & \ldots & x_{p1} \\ x_{12} & x_{22} & \ldots & x_{p2} \\ \vdots & \vdots & \vdots & \vdots \\ x_{1v} & x_{2v} & \ldots & x_{pv} \\ \vdots & \vdots & \vdots & \vdots \\ x_{1N} & x_{2N} & \ldots & x_{pN} \end{bmatrix}.$$

Prior to calculating any distance metric used to select training and test sets, the data is standardized and orthonormalized to reduce overall sensitivity due to factor ranges and orientation. A standardization step is applied to the elements of the

candidate design matrix, X where

$$\begin{aligned} x_{iv} &= (X_{iv} - X_{i\cdot}) \,/\, \left[\sum (X_{iv} - X_{i\cdot})^2\right]^{1/2} \\ \text{where} \quad X_{i\cdot} &= \sum_v X_{iv}/N \end{aligned} \tag{2.9}$$

and X_{iv} are the raw coordinate values from the candidate matrix.

Next, the data is orthonormalized. The candidate design matrix is decomposed using a Choleski variant of Gaussian elimination by

$$X'X \to T'T \tag{2.10}$$

where T is upper triangular and X is assumed to be of rank p. Finally, the candidate design is transformed by

$$W = XT^{-1} \tag{2.11}$$

with $W'W = I_p$ where I_p is a square $p \times p$ identity matrix. The experimental design is sequentially selected from the elements of the orthonormal candidate matrix with candidate points, $W = w_1, w_2, \ldots, w_N$. Let $Q = q_1, q_2, \ldots, q_n$ and $R = r_1, r_2, \ldots, r_n$ represent the training and test sets respectively. Therefore, $Q \cup R \subseteq W$ and $Q \cap R = \emptyset$. In the absence of a set of starting points, the first two points included in the training set are selected by calculating

$$\begin{aligned} \{u^*, v^*\} &= \max_{v<u} \|w_v - w_u\|^2 \\ &= \sum_{k=1}^{p} (w_{kv} - w_{ku})^2 \end{aligned} \tag{2.12}$$

which identifies the two most separated points as the first included in the training set, $Q = \{w_{u^*}, w_{v^*}\}$. The points are then removed from the list of candidate points,

$W = W \backslash \{w_{u^*}, w_{v^*}\}$ where elements to the right of $\backslash$ are removed from the set W. In the rare case that there is not a unique solution to Equation (2.12), ties are broken based on the pair with the smallest index, v. Next, training set points are sequentially selected by defining the squared distance from point v to point u as

$$D_{vu}^2 = \|w_v - w_u\|^2 \tag{2.13}$$

Letting $Q = q_1, q_2, \ldots, q_i, \ldots, q_k$ for $k < n$ represent the k points already included in the training set, the $k+1$st design point is chosen as follows. Let

$$\Delta_v^2(k) = \min_{i \in Q} \{D_{1v}^2, D_{2v}^2, \ldots, D_{kv}^2\} \tag{2.14}$$

for $v \in W$ be the squared distance from the point v (not yet in the design) to the nearest design point already included. Selection of the $k+1$st design point is performed by choosing the point remaining in the $(N-k)$ candidate points which is farthest from an existing design point using the criterion

$$\Delta_{k+1}^2 = \max_{v \notin Q} \{\Delta_v^2(k)\}. \tag{2.15}$$

Assuming n was chosen as $n = \frac{1}{2}N$, once n points were included in the training set, the remainder of the points are placed in the test set.

2.3.2 DUPLEX.

While the CADEX algorithm focuses strictly on the training set, the DUPLEX algorithm attempts to create training and test sets covering similar areas of the factor space and having similar statistical properties. As in the CADEX algorithm, candidate points are first standardized and orthonormalized as in equations (2.9)-(2.11)

producing W. The Euclidean distance between all possible pairs of points (u, v) is calculated using Equation (2.13). The distance between all pairs (u, v) need only be calculated once.

To begin the algorithm, the points $\{v^*, u^*\}$ satisfying Equation (2.12) are again included in the training set, $Q = \{w_{u^*}, w_{v^*}\}$ thus placing the two most separated points in the training set; the points are again removed from the candidate list, $W = W \backslash \{w_{u^*}, w_{v^*}\}$. Equation (2.12) is once again solved for the remaining candidate points in W and the next most separated points, $\{u^*, v^*\}$, are placed in the test set, $R = \{w_{u^*}, w_{v^*}\}$ while the candidate points $\{v^*, u^*\}$ are once again removed from the candidate list, $W = W \backslash \{w_{u^*}, w_{v^*}\}$. The remainder of the candidate points are placed in alternating fashion in the training and test sets based on their distance from points already in the specified set. Let s be the current algorithm iteration which is at $s = 3$ after initializing the training and test sets. Then when s is odd, letting $q_1, q_2, \ldots, q_i, \ldots, q_k$ for $k < n$ represent the k points already included in the training set, the $k + 1$st training set design point is chosen as follows. First, the minimum distance from a point not in either the training or test set to the nearest training set point is defined as

$$\Delta_v^2(k) = \min_{i \in Q} \{D_{1v}^2, D_{2v}^2, \ldots, D_{kv}^2\} \tag{2.16}$$

for $v \in W$. The $k + 1$st training set point is then selected from the $N - k$ candidate points by using the criterion

$$\Delta_{k+1}^2 = \max_{v \notin Q, v \notin R} \{\Delta_v^2(k)\}. \tag{2.17}$$

The k+1st point is then removed from the candidate list, $W = W \backslash w_{k+1}$. Similarly, when s is even, letting $r_1, r_2, \ldots, r_j, \ldots, r_g$ for $g < n$ represent the g points already included in the test set, the $g + 1$st test set design point is chosen as follows. First,

let

$$\Delta_u^2(g) = \min_{j \in R} \{D_{1u}^2, D_{2u}^2, \ldots, D_{gu}^2\} \tag{2.18}$$

represent the candidate point closest to a point in the test set, u, for $u \in W$. The $g+1$st test set point is then selected from the $N-g$ candidate points by using the criterion

$$\Delta_{g+1}^2 = \max_{u \notin Q, u \notin R} \{\Delta_u^2(g)\}. \tag{2.19}$$

The $g+1$st point is then removed from the candidate list, $W = W \backslash w_{g+1}$. This process is continued until all $n \leq N$ candidate points are added to either the training or test set.

2.3.3 Characterizing Noise.

A hyperspectral image, often referred to as an image cube, consists of p spectral bands of an $m \times n$ spatial pixel representation of a sensed area. Each pixel in the spectral dimension represents an intensity of energy reflected back to the sensor. There are several potential observable noise characteristics that are used to define the noise present in a hyperspectral image. Mindrup *et al.* [57] focused on three: the Fisher ratio, the ratio of target pixels and the number of clusters.

The Fisher ratio, $\mathbf{z}_1$, described by Duda *et al.* [25, 55] is a measure for the discriminating power of a variable. The Fisher ratio for the i^{th} individual image, $i = 1, 2, \ldots, I$ where I is the total number of images under consideration, is defined as the average Fishers ratio across each image band, $k = 1, 2, \ldots, K$. Thus, the Fisher ratio for image i is

$$z_{i1} = \frac{\sum_{k=1}^{K} \left(\frac{\left(\mu_{a_{i,k}} - \mu_{b_{i,k}}\right)^2}{\sigma_{a_{i,k}}^2 + \sigma_{b_{i,k}}^2} \right)}{K} \tag{2.20}$$

where $\mu_{a_{i,k}}$ and $\sigma_{a_{i,k}}^2$ are the mean and variance of the anomalous pixels, a, in band k

of image i and $\mu_{b_{i,k}}$ and $\sigma^2_{b_{i,k}}$ are the mean and variance of the background pixels, b, in band k of image i, all defined from a truth mask.

The ratio of target pixels, $\mathbf{z}_2$, was calculated if there was a truth mask for each image, $i = 1, 2, \ldots, I$, by

$$z_{i2} = \frac{v_i}{b_i} \tag{2.21}$$

where v_i and b_i represent the number of anomalous pixels and background pixels in image i respectively.

The number of clusters represents the number of homogenous groups of pixels within an image. The number of clusters, $\mathbf{z}_3$, was recorded for each image, $i = 1, 2, \ldots, I$ using X-means as developed by Pelleg and Moore [66].

Each noise feature vector was standardized by

$$\hat{\mathbf{z}}_k = \frac{\mathbf{z}_k - \mu_{\mathbf{z}_k}}{\sigma_{\mathbf{z}_k}} \tag{2.22}$$

where $\mu_{\mathbf{z}_k}$ and $\sigma_{\mathbf{z}_k}$ represent the mean and standard deviation of the k^{th} noise vector, $\mathbf{z}_k$. The three standardized noise feature vectors were combined in an $I \times q$ noise matrix, $Z = [\hat{\mathbf{z}}_1 \quad \hat{\mathbf{z}}_2 \quad \hat{\mathbf{z}}_3]$, with I total images and $q = 3$ noise variables.

2.3.4 SSTATS Method - Preliminaries.

For this research the number of images, I, are assumed even and split equally between the training and test sets. For the cost functions described below, scores for each set were added together yielding $\binom{I}{I/2}/2 = n$ unique couplets of training and test sets. Let $(\mathcal{S}_w, \overline{\mathcal{S}}_w)$ represent the w^{th} couplet; here $\mathcal{S}_w$ is the training set and $\overline{\mathcal{S}}_w$ is the test set. Then the set of unique couplets is $\mathcal{S} = \big((\mathcal{S}_1, \overline{\mathcal{S}}_1), (\mathcal{S}_2, \overline{\mathcal{S}}_2), \ldots, (\mathcal{S}_n, \overline{\mathcal{S}}_n)\big)$.

The training (tr) and test (te) set selection problem can be abstracted to be

$$(\mathcal{S}_{tr}, \mathcal{S}_{te}) = \left(\mathcal{S}_{w^*}, \overline{\mathcal{S}}_{w^*}\right) = \arg\max_{w} f\left((\mathcal{S}_w, \overline{\mathcal{S}}_w)\right) \tag{2.23}$$

for an appropriate cost function, f.

The training set of images are used in RPD to approximate the true anomaly detection function in Equation (2.5) by

$$\hat{G}(x|z = z_{tr}) = \hat{\beta}_0 + x'\hat{\beta} + x'\hat{B}x + z'\hat{\gamma} + x'\hat{\Delta}z + \epsilon \tag{2.24}$$

where x are the anomaly detection algorithm settings and z_{tr} are the noise features collected from a set of training images, $\mathcal{S}_{tr} \subset \mathcal{S}$. The test images are used to assess the efficacy of the fitted model as well as the representativeness of the selected training set.

2.3.5 SSTATS Method.

Let $Z_{\mathcal{S}_w}$ and $Z_{\overline{\mathcal{S}}_w}$ represent the standardized noise matrices for a given couplet $(\mathcal{S}_w, \overline{\mathcal{S}}_w)$. The standardized noise matrices, $Z_{\mathcal{S}_w}$ and $Z_{\overline{\mathcal{S}}_w}$, were incorporated in an objective function designed to separate the images relative to the noise space. Herein an objective function is proposed to maximize the volume of both the training and test sets while maintaining an acceptable separation between individual points within both the training and test sets, respectively. The objective function is computed in terms of two set separation distance measures. The first is the average Euclidean distance from each training or test set point to its respective mean vector; the second considers the average distance between points within the training and test sets. A D-optimal score is used to compare the volumes of these sets and the set with the larger volume is identified as the training set.

Similar problems have been studied in the area of designed experiments. The D-optimal criterion is used to select designs that minimize the generalized variance and maximize the volume of the convex hull of $X'X$, sometimes referred to as the information matrix where X is the experimental design matrix, thereby minimizing the confidence region for the regression coefficients [78]. A D-optimal design maximizes

$$D = \frac{|X'X|}{K^p} \tag{2.25}$$

where K is the number of experimental design points, p is the number of parameters in the model and $|X'X|$ is the determinant of the information matrix.

When considering training and test set combinations from a discrete number of possible images, I, a D optimal score was calculated for both sets. The D optimal criterion of the first set for any couplet $w = 1, 2, \ldots, W$ is

$$D_{\mathcal{S}_w} = \frac{|Z_{\mathcal{S}_w}{}' Z_{\mathcal{S}_w}|}{K^p}. \tag{2.26}$$

The D optimal criterion for the complement set in couplet $w = 1, 2, \ldots, W$ was found by replacing $\mathcal{S}_w$ in Equation (2.26) with $\overline{\mathcal{S}}_w$.

Another expression reflecting the spread of the noise variables is defined by the average Euclidean distance from each training or test set point to its respective mean vector. The average distance between each image in the first set, $i \in \mathcal{S}_w$, and the mean vector for all noise variables in the first set, $\boldsymbol{m}_{\mathcal{S}_w}$, for a couplet $w = 1, 2, \ldots, W$ is defined as

$$\delta_{\mathcal{S}_w} = \frac{\sum_{i \in \mathcal{S}_w} \left((Z_i' - \boldsymbol{m}_{\mathcal{S}_w})' (Z_i' - \boldsymbol{m}_{\mathcal{S}_w}) \right)^{\frac{1}{2}}}{I/2} \tag{2.27}$$

where Z_i represents row $i \in \mathcal{S}_w$ of the noise matrix, Z. A similar expression was used for images in the second set, $i \in \overline{\mathcal{S}}_w$.

As a final expression reflecting the spread of the noise variables, the average dis-

tance between points within the two sets was considered. The average distance between points within both sets was calculated using the Euclidean distance. The average separation distance in the first set for couplet $w = 1, 2, \ldots, W$ is

$$d_{\mathcal{S}_w} = \frac{\sum_{i \in \mathcal{S}_w} \sum_{j > i \in \mathcal{S}_w} \left(\left(Z_i' - Z_j' \right)' \left(Z_i' - Z_j' \right) \right)^{\frac{1}{2}}}{\binom{I/2}{2}} \tag{2.28}$$

where the denominator represents the total number of pairs of images being considered and Z_i and Z_j represent the noise values for images i and j found on rows $j > i \in \mathcal{S}_w$ of the noise matrix, Z, respectively. A similar expression was used for images in the second set, $j > i \in \overline{\mathcal{S}}_w$.

D_{S_w}, δ_{S_w} and d_{S_w} and their complements were calculated for each possible unique couplet of images, $w = 1, 2, \ldots, W$. Next, the scores were standardized using Equation (2.22) to give them all equal weighting. Finally, an objective function was defined to characterize image noise based on these different standardized volume and separation differences. The objective function with respect to a specified couplet, $(\mathcal{S}_w, \bar{\mathcal{S}}_w)$, is:

$$f(\mathcal{S}_w, \overline{\mathcal{S}}_w) \;=\; \frac{\hat{d}_{\mathcal{S}_w} + \hat{d}_{\overline{\mathcal{S}}_w}}{1 + |\hat{d}_{\mathcal{S}_w} - \hat{d}_{\overline{\mathcal{S}}_w}|} + \frac{\hat{\delta}_{\mathcal{S}_w} + \hat{\delta}_{\overline{\mathcal{S}}_w}}{1 + |\hat{\delta}_{\mathcal{S}_w} - \hat{\delta}_{\overline{\mathcal{S}}_w}|}. \tag{2.29}$$

Based on Equation (2.23), the optimal couplet, $(\mathcal{S}_{w^*}, \overline{\mathcal{S}}_{w^*})$, is

$$\left(\mathcal{S}_{w^*}, \overline{\mathcal{S}}_{w^*} \right) = \arg\max_{w} f\left(\mathcal{S}_w, \overline{\mathcal{S}}_w \right). \tag{2.30}$$

Within this optimal couplet, the set with the largest D-optimal criterion value was selected as the optimal training set, tr^*. The remaining set was identified as the

optimal test set, te^*.

$$\begin{cases} \text{if } D_{\mathcal{S}_{w^*}} > D_{\overline{\mathcal{S}}_{w^*}} & tr^* = \mathcal{S}_{w^*} \quad \text{and} \quad te^* = \overline{\mathcal{S}}_{w^*} \\ \text{otherwise} & tr^* = \overline{\mathcal{S}}_{w^*} \quad \text{and} \quad te^* = \mathcal{S}_{w^*} \end{cases} \tag{2.31}$$

2.4 Simulation Experiment

In what follows, a simulation meta-experiment is described that compares the performance of the data splitting algorithms: CADEX, DUPLEX, SSTATS and random selection. The meta-experiment consists of a basic experiment (represented by Blocks shaded gray in Figure 2.3) replicated $m = 1000$ times (represented by white Blocks in Figure 2.3). Individual Blocks in Figure 2.3 are numbered and will be referenced as such. A broad overview of the meta-experiment is described below. Then, a more detailed description follows in Sections 2.4.1-2.4.5.

In the absence of a true anomaly detector function for generating responses, a simulated "truth" model, $y = G(x, z)+\epsilon$, was created and optimal settings were identified in Blocks 1 and 2 of Figure 2.3. Noise was generated to match the noise distributions observed from Hyperspectral Digital Imagery Collection Equipment (HYDICE) sensor Forest Radiance I and Desert Radiance II collection events in Block 3. To allow a graphical comparison of training and test sets, two control and two noise factors were used in the experiment. "Optimal" training sets were selected using SSTATS, CADEX and DUPLEX as well as randomly selected sets in Block 4 of Figure 2.3. Then, responses from the simulated truth model were generated for an RPD of the control variables and training set noise in Block 5. Stepwise regression was performed on the experimental design and response vector to identify estimates for the RPD model coefficients yielding the RPD function, $\hat{y} = \hat{G}(x, z) + \epsilon$ in Block 6. In Block 7, RPD optimal control settings identified from the estimated model led to approximated optimal control settings. Next, training and test set points were

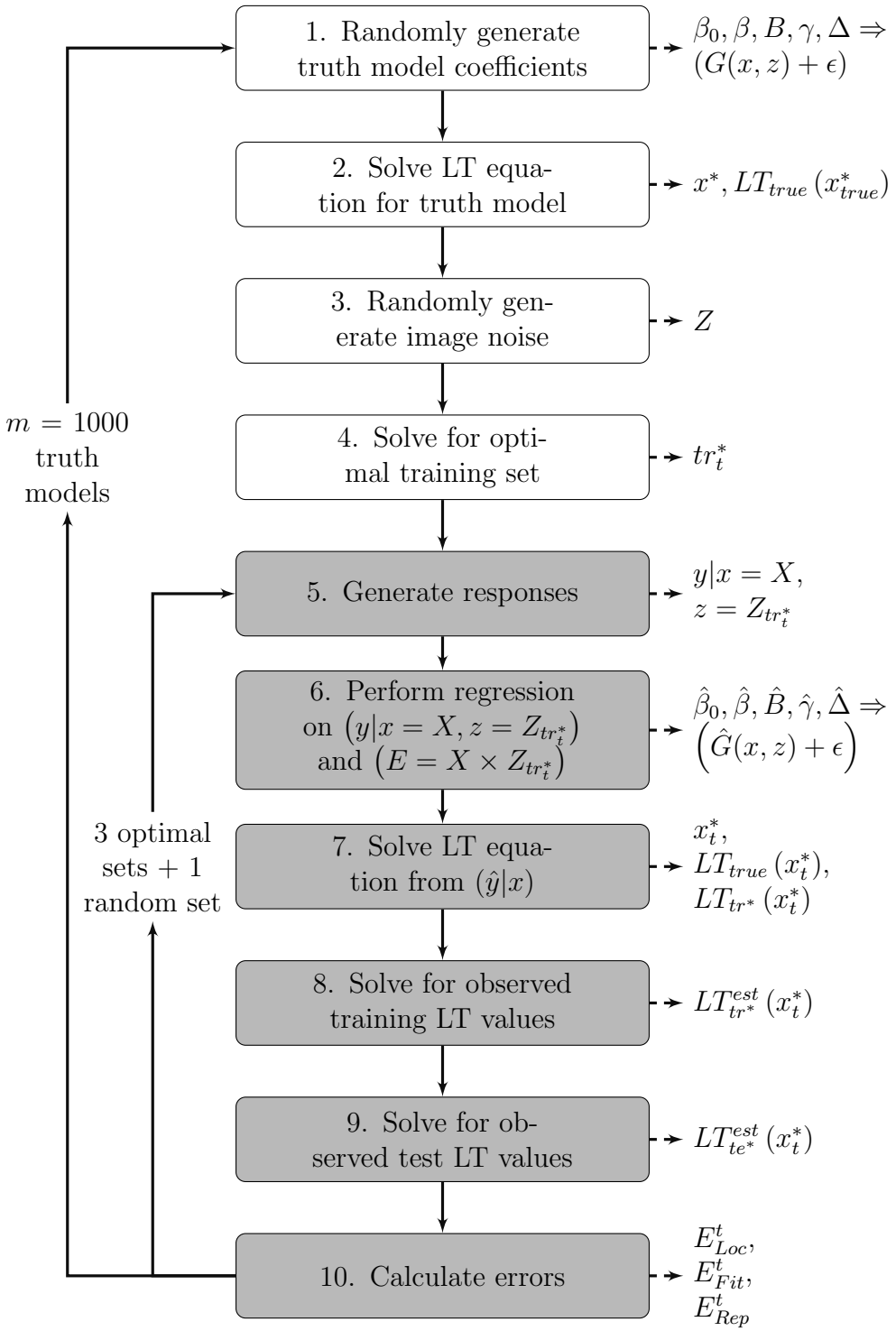


Figure 2.3. Simulation experiment and error estimation.

used to assess the representativeness of the training set in Blocks 8 and 9 of Figure 2.3. Finally, errors associated with the location of the optimal settings, the fit of the RPD model and the representativeness of the training and test sets were defined to assess the selection strategy performances as well as the performance of the randomly selected training set in Block 10.

2.4.1 Develop Truth Model/Identify Optimal Settings.

In Blocks 1 and 2 of the meta-experiment in Figure 2.3, a truth model was created and optimal settings were identified. The truth model coefficients, $\beta_0, \beta, B, \gamma$ and Δ, were initially taken from the fitted model in Myers *et al.* [63, pg. 567]. Standard normal ($N(0,1)$) random variates were added to each coefficient for a given iteration producing variability from model to model. The true process model became $y = G(x,z) + \epsilon$, where $\epsilon \sim N(0, \sigma^2 = 2)$. There were $m = 1000$ different truth models created to observe variability across different truth models. Once true parameters were selected, the true optimal control variable settings, x^*, and optimal LT value, $LT_{true}\left(x^*_{true}\right) = LT(y|x = x^*)$, were identified using Equation (2.8).

2.4.2 Create Image Noise/Identify Optimal Training Sets.

Noise was generated in Block 3 of Figure 2.3 representing Fisher's score and percent targets based on fitted distributions of eight images from the HYDICE Desert and Forest Radiance data sets. Each image was halved to double the total number of images to 16. For the most part, noise characteristics for the upper and lower half of an image were homogenous. Figure 2.4 shows the process for creating two image halves with noise vectors from one HYDICE image.

The noise features from the HYDICE images are arrayed in Table 2.1. Following the process depicted in Figure 2.4, each original image was halved (1 -upper half, 2 -lower half) and renamed with a new image identification.

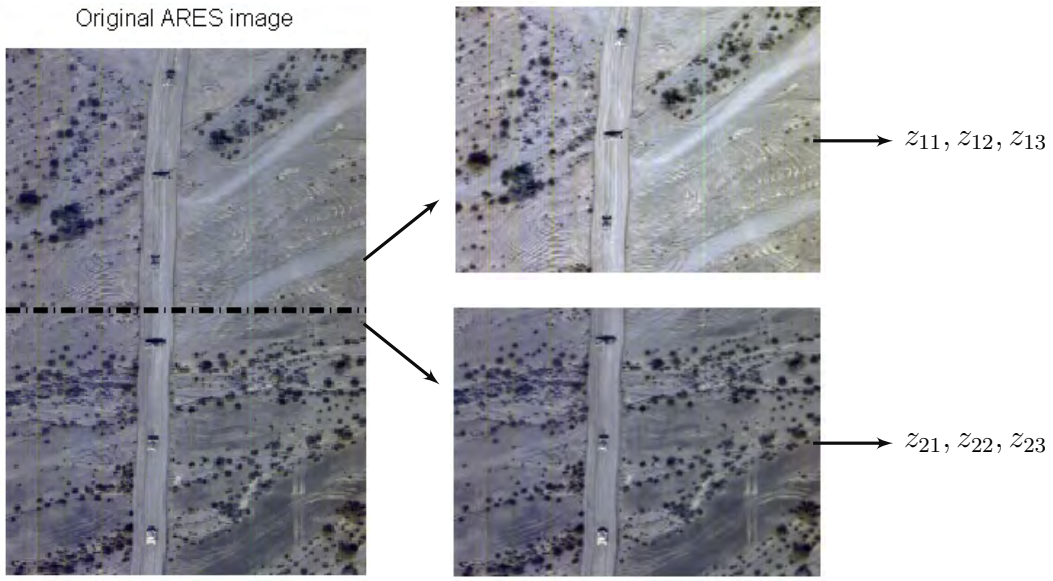


Figure 2.4. Image noise characterization.

Table 2.1. Observed image noise characteristics.

Original Image	Image Half	New Image ID	Fishers Score ($\mathbf{Z}_1$)	Percent of Targets ($\mathbf{Z}_2$)	Number of Clusters ($\mathbf{z}_3$)
1D	Upper	1	1.780	0.004	3
1D	Lower	2	1.627	0.003	3
1F	Upper	3	0.433	0.039	6
1F	Lower	4	0.315	0.022	10
2D	Upper	5	0.096	0.025	3
2D	Lower	6	0.176	0.029	3
2F	Upper	7	0.963	0.008	8
2F	Lower	8	0.931	0.009	7
3D	Upper	9	0.169	0.003	3
3D	Lower	10	1.430	0.003	3
3F	Upper	11	0.265	0.005	5
3F	Lower	12	0.215	0.008	5
4F	Upper	13	0.083	0.005	7
4F	Lower	14	0.078	0.006	8
4	Upper	15	1.409	0.016	7
4	Lower	16	2.638	0.028	4

The HYDICE image noise was fit to standard probability distributions using the ARENA input analyzer. The Fisher's score was fit to a Beta distribution with $\alpha = 0.511$ and $\beta = 1.38$ with an additive shift parameter of 2.9. The percent targets was fit to an exponential distribution with a mean of 0.0123 [38]. These distributions were used to create random image noise in the simulation study. There were 16 training points generated for each truth model based on the random noise distributions, z. This meant there were $n = \binom{16}{8}/2$ or 6435 total possible unique couplets of training and test sets. The random noise data was standardized and the noise vectors were combined to form $Z = [\hat{\mathbf{z}}_1 \quad \hat{\mathbf{z}}_2]$.

Next, in Block 4 of the meta-experiment in Figure 2.3, training sets were selected using CADEX, DUPLEX and SSTATS as well as a randomly selected training set yielding tr_C^*, tr_D^*, tr_S^* and tr_R^* respectively. The associated test sets were te_C^*, te_D^*, te_S^* and te_R^*. The training and test subsets were used to perform RPD.

2.4.3 Perform RPD.

Once the training sets were selected, an experimental design for RPD, E_t for $t = \{C, D, S, R\}$, was developed. A 3^2 factorial design with one replicate was used in the initial orthogonal design for the control variables, X. The orthogonal design, X, was augmented with every row of training noise variables to create E_t:

$$E_t = X \times Z_{tr^*} \tag{2.32}$$

where $\times$ represents the Cartesian product. Initial responses were generated for each basic experiment in Block 5 of Figure 2.3 by substituting values for each row of E_t into Equation (2.5) along with random process variance drawn from $\epsilon \sim N(0, \sigma^2 = 2)$ yielding $y|x = X, z = Z_{tr_t^*}$. Figure 2.3 reflects a change in shade at Block 5 to denote the beginning of the basic experiment.

Next, Block 6 of Figure 2.3 depicts stepwise regression performed on each vector

of training set responses and the associated experimental design, $y|x = X, z = Z_{tr_t^*}$ and E_t for $t = \{C, D, S, R\}$ yielding $\hat{\beta_{0t}}, \hat{\beta}_t, \hat{B}_t, \hat{\gamma}_t$ and $\hat{\Delta}_t$. These parameter estimates were used in Block 7 of Figure 2.3 to identify the estimated optimal control settings based on the optimal training set, x_t^*, using Equation (2.4). The LT score (Lin and Tu "target is best" MSE with $T = 5$) in the truth surface evaluated at $x = x_t^*$, $LT_{true}(x_t^*)$, was found by using the true parameters ($\beta_0, \beta, B, \gamma, \Delta$ and σ^2) in

$$\begin{aligned} LT_{true}(x_t^*) &= LT(y|x = x_t^*) \\ &= \left(\beta_0 + x_t^{*\prime}\beta + x_t^{*\prime}Bx_t^* - T\right)^2 \\ &+ \sigma_z^2\left(\gamma' + x_t^{*\prime}\Delta\right)\left(\gamma' + x_t^{*\prime}\Delta\right)' + \sigma^2. \end{aligned} \tag{2.33}$$

This value was used to assess "fit" error (described in Section 2.4.5) by comparing the optimal LT value, $LT_{true}(x_{true}^*)$, with the estimated LT value in the true surface, $LT_{true}(y|x_t^*)$. For a simple example, consider the linear truth model with a single control and noise variable, $Z \sim N(0, 1)$, as specified below.

$$y = G(x, z) = 1.25 + 0.55x - 0.68xz - 0.2z \tag{2.34}$$

Assuming the target mean value is 2.0, the true LT model becomes

$$LT_{true} = (1.25 + 0.55x - 2)^2 + (-0.2 - 0.68x)^2 + 1 \tag{2.35}$$

The optimal control setting would be $x_{true}^* = 0.36$ and the optimal LT value becomes $LT_{true}(x_{true}^* = 0.36) = 1.5$. Further, let the fitted model from regression of a training set be

$$\hat{y} = \hat{G}(x, z) = 1.2 + 0.49x - 0.59xz - 0.22z. \tag{2.36}$$

The estimated LT model is

$$LT_{tr^*} = (1.2 + 0.49x - 2)^2 + (-0.22 - 0.59x)^2 + 1. \tag{2.37}$$

The estimated optimal control setting becomes $x_t^* = 0.45$ with an estimated LT value of $LT_{tr^*}\left(x_t^* = 0.45\right) = 1.57$. The LT score in the truth surface evaluated at $x = x_t^*$ becomes $LT_{true}\left(x_t^* = 0.45\right) = 1.51$.

2.4.4 Training and Test Image LT.

LT scores for training points, $z_{tr} = \{0.75, -0.5\}$, and test points, $z_{te} = \{0.25, -0.45\}$, were calculated to consider the representativeness of the training and test sets as shown in Blocks 8 and 9 of Figure 2.3. As in practice, RPD optimal control settings identified from training points would be applied to the test points to assess setting adequacy. All test responses would then used to calculate an estimated LT score. LT scores for test set $t = \{C, D, S, R\}$ at the estimated optimal control settings, $LT_{te}^{est}\left(x_t^*\right)$, were found by calculating the mean, $\bar{y}_{te}$ and variance, s_{te}^2, across all test responses and solving Equation (2.38) for a given target value, T.

$$LT_{te}^{est}\left(x_t^*\right) = \left\{\bar{y}_{te} - T\right\}^2 + s_{te}^2. \tag{2.38}$$

In order to have a one-to-one comparison of the sets, the same process was applied to the training points for the training set mean, $\bar{y}_{tr}$, and training set variance, s_{tr}^2. The LT score for training set, t, becomes

$$LT_{tr}^{est}\left(x_t^*\right) = \left\{\bar{y}_{tr} - T\right\}^2 + s_{tr}^2. \tag{2.39}$$

Returning to the simple example problem with a single control and noise variable, consider four new responses divided into training and test. The training image LT score is

$$LT_{tr}^{est}\left(x_t^*\right) = (1.43 - 2)^2 + 0.2 = 0.52. \tag{2.40}$$

Similarly, the test image LT score is

$$LT_{te}^{est}\left(x_t^*\right) = (1.55 - 2)^2 + 0.06 = 0.27. \tag{2.41}$$

2.4.5 Error Definitions.

In Block 10 of Figure 2.3, three errors were used to describe the performance for a given training and test set. First, a measure was defined to compare the true location of x^* to the optimal control settings from the regression model, x_t^*. The location error for a given training set $t = \{C, D, S, R\}$, MSE_{Loc}^t, is measured as the Euclidean distance from the estimated optimal control settings to the true optimum:

$$E_{Loc}^t = RMSE_{Loc}^t = \left((x^* - x_t^*)'(x^* - x_t^*)\right)^{1/2}. \tag{2.42}$$

Next, an error to describe the difference between the optimum LT score, $LT_{true}\left(x_{true}^*\right)$, and the regression model optimum LT score for a given training set, $LT_{true}\left(x_t^*\right)$, was developed. This value represented the absolute fit error for the model created using training set $t = \{C, D, S, R\}$:

$$E_{Fit}^t = \Delta MSE_{Fit}^t = \left|LT_{true}\left(x_{true}^*\right) - LT_{true}\left(x_t^*\right)\right|. \tag{2.43}$$

Finally, an error estimating the representativeness of the training and test sets was defined comparing the absolute difference between $LT_{tr}^{est}\left(x_t^*\right)$ and $LT_{te}^{est}\left(x_t^*\right)$

$$E_{Rep}^t = \Delta MSE_{Rep}^t = \left|LT_{tr}^{est}\left(x_t^*\right) - LT_{te}^{est}\left(x_t^*\right)\right|. \tag{2.44}$$

Returning to the one control factor, one noise variable example from Section 2.4.3, the location error becomes

$$E_{Loc}^t = \left((0.36 - 0.45)^2\right)^{1/2} = 0.09. \tag{2.45}$$

The fit error is calculated as

$$E_{Fit}^t = |1.5 - 1.51| = 0.01. \tag{2.46}$$

Finally, the representative error for the example is

$$E^t_{Rep} = |0.52 - 0.27| = 0.25. \tag{2.47}$$

Table 2.2 summarizes the different example LT scores and the data they were computed from.

Table 2.2. Example LT table.

	Truth	**Train**		**Test**	
x	0.36	0.45	0.45	0.45	0.45
z	-	0.75	-0.5	0.25	-0.45
y	-	1.12	1.75	1.37	1.73
$\bar{y}$	-	1.80		0.70	
s^2	-	1.55		0.46	
$LT_{true}(x^*)$	1.5	-		-	
$LT_{true}(x^*_t)$	-	1.51		-	
$LT_{tr^*}(x^*_t)$	-	1.57		-	
$LT^{est}_{tr^*}(x^*_t)$	-	1.59		-	
$LT^{est}_{te^*}(x^*_t)$	-	-		2.15	

Figure 2.5 gives another illustration of the different LT values and errors described to this point. In the figure, points 1 and 2 represent training and test LT values observed at the estimated optimum point, $LT^{est}_{tr}(x^*_t)$ and $LT^{est}_{te}(x^*_t)$ respectively.

2.4.6 Simulation Results.

The meta-experiment in Figure 2.3 was performed 1000 times. Location, fit and representative errors were calculated for all four training sets, $t = C, D, S, R$. This allowed a comparison between the proposed methodology and response surfaces generated from the other training sets. Thus, the simulation could be considered a Binomial experiment made up of 1000 independent identical Bernoulli trials. Each independent Bernoulli trial would measure whether the training set selected by the proposed cost function resulted in a smaller error than one from another training set. For instance, when comparing SSTATS with a randomly selected training set, if the

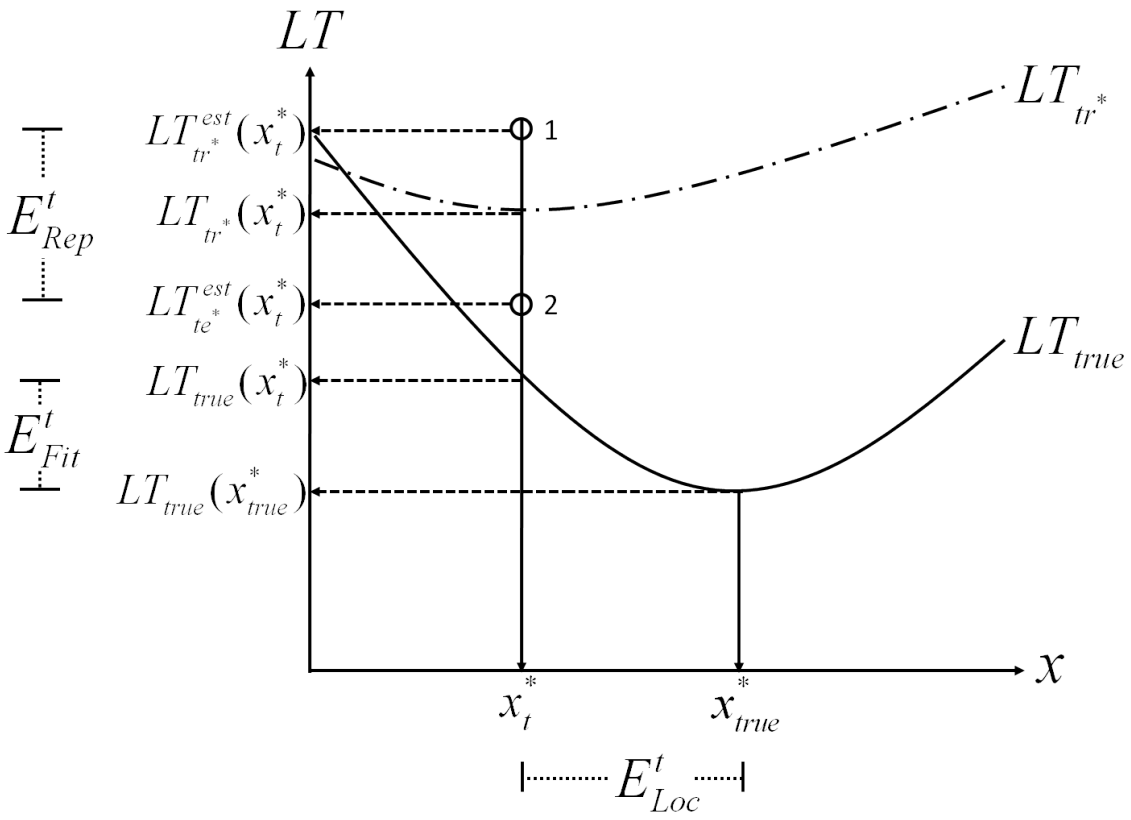


Figure 2.5. Summary figure of errors and associated points.

location error from the SSTATS training set was less than the location error using a random training set ($E^S_{Loc} < E^R_{Loc}$), then the trial was a success. The probability of success, $\hat{p}$ was the ratio of the number of successes out of 1000 independent trials where $\hat{p}_j = Pr\left(E^S_j < E^{t'}_j\right)$, $j \in \{Loc, Fit, Rep\}$, $t' = C, D, R$. Thus, a confidence interval for p could be formed comparing each training set type by error. Confidence intervals with lower limits greater than $p = 0.5$ provide evidence that the training set chosen by the SSTATS objective function yielded smaller errors than training sets selected using one of the other training set selection methods.

Figure 2.6 gives 95% confidence intervals for the location error, fit error and representative error probabilities, $\hat{p}$, comparing SSTATS with a randomly selected training set. The location, fit and representative errors had confidence intervals on $\hat{p}$ that did not include $p = 0.5$ implying a significant difference between the errors associated with a random training set and a SSTATS training set. This demonstrated the benefit of choosing training and test sets of images with an adequate separation of noise variables using SSTATS rather than randomly picking training and test sets of images. This separation leads to more consistent results on test sets points compared with a random training set.

Figure 2.7 gives 95% confidence intervals comparing SSTATS with training sets selected using CADEX. The location, fit and representative errors had confidence intervals on $\hat{p}$ that did not include $p = 0.5$ implying a significant difference between the errors associated with a CADEX training set and a SSTATS training set with SSTATS outperforming CADEX.

Figure 2.8 gives 95% confidence intervals comparing SSTATS with training sets selected using DUPLEX. The location and fit errors had confidence intervals on $\hat{p}$ that did not include $p = 0.5$ implying a significant difference between the errors associated with a DUPLEX training set and a SSTATS training set. The representative errors

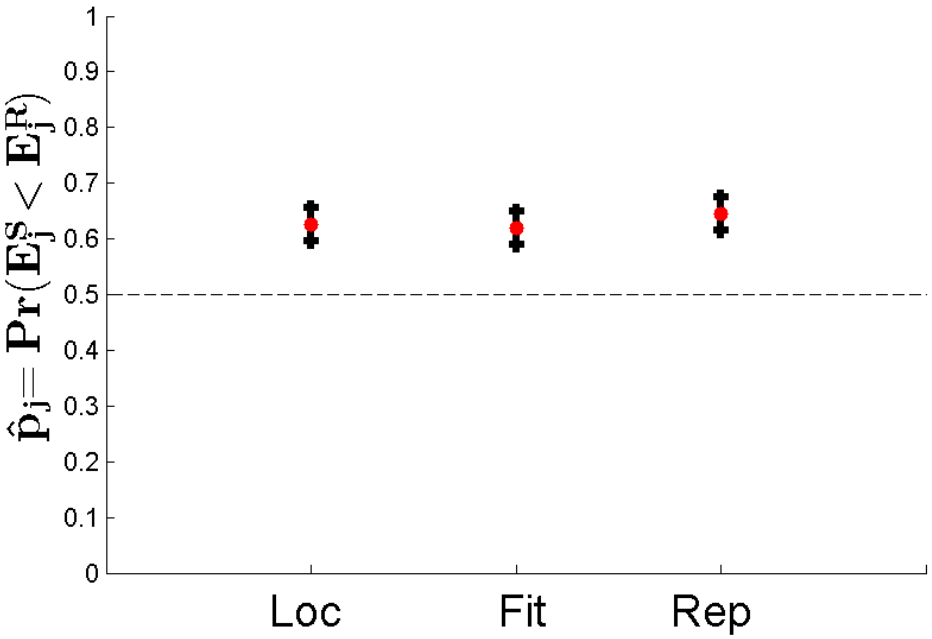


Figure 2.6. SSTATS vs. random confidence intervals.

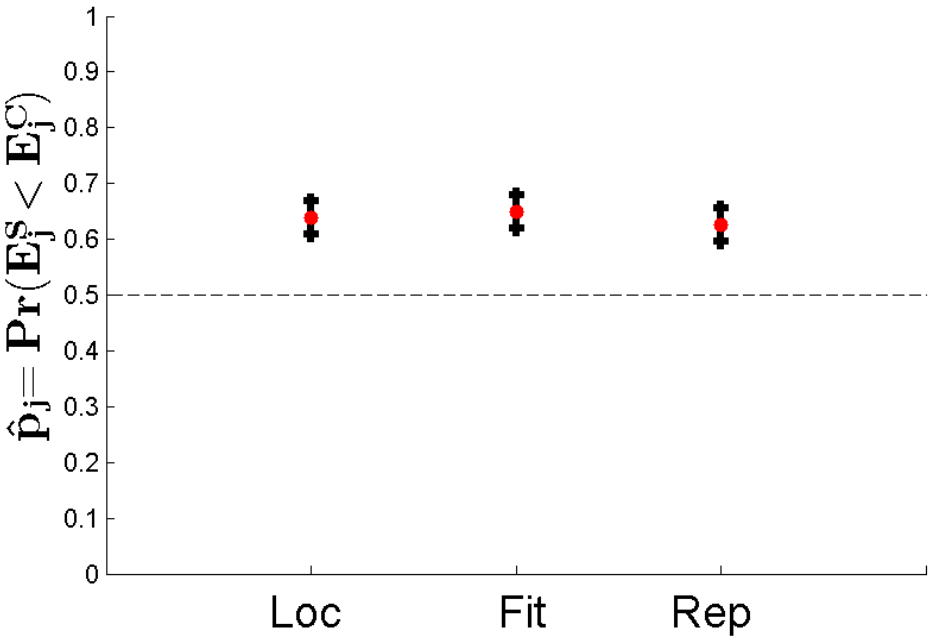


Figure 2.7. SSTATS vs. CADEX confidence intervals.

were not statistically different from a SSTATS training set and a DUPLEX training set. This was not surprising as both methods attempt to identify representative training and test subsets. Overall, SSTATS outperformed DUPLEX in terms of model fits showing the power of the proposed methodology.

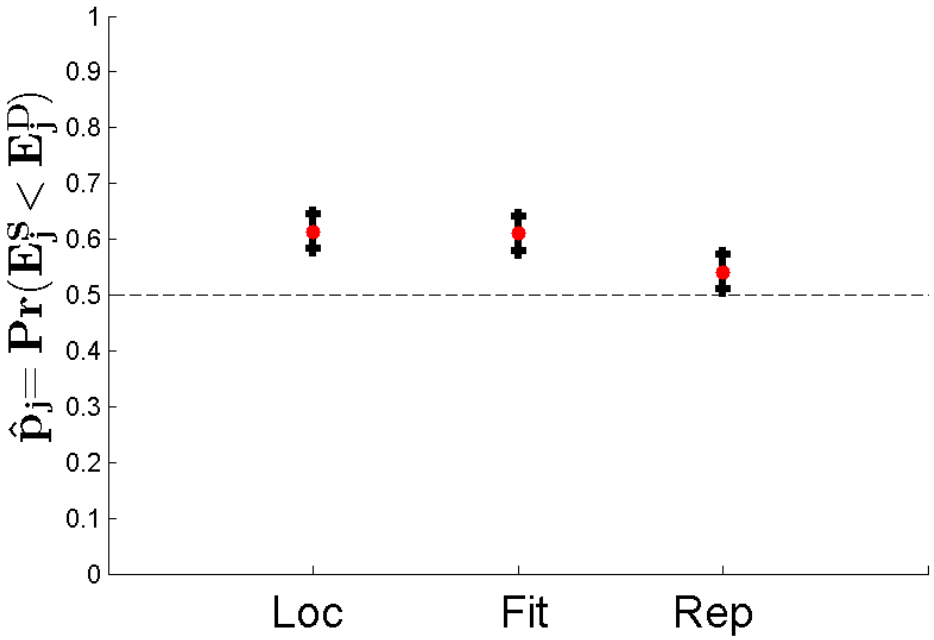


Figure 2.8. SSTATS vs. DUPLEX confidence intervals.

Further evidence of improved performance using a training set selected using SSTATS rather than DUPLEX was gleaned from the maximum difference in errors between DUPLEX training sets and SSTATS training sets. The error differences were calculated by $E^S_{Rep} - E^D_{Rep}$. Large negative values show increased errors from the DUPLEX set and were preferred while large positive values reflect larger magnitudes of errors from SSTATS training sets. Histograms were used to further illuminate the distribution of error differences. Figure 2.9 displays the representative error histogram.

On the average, there was no significant difference in representative error between

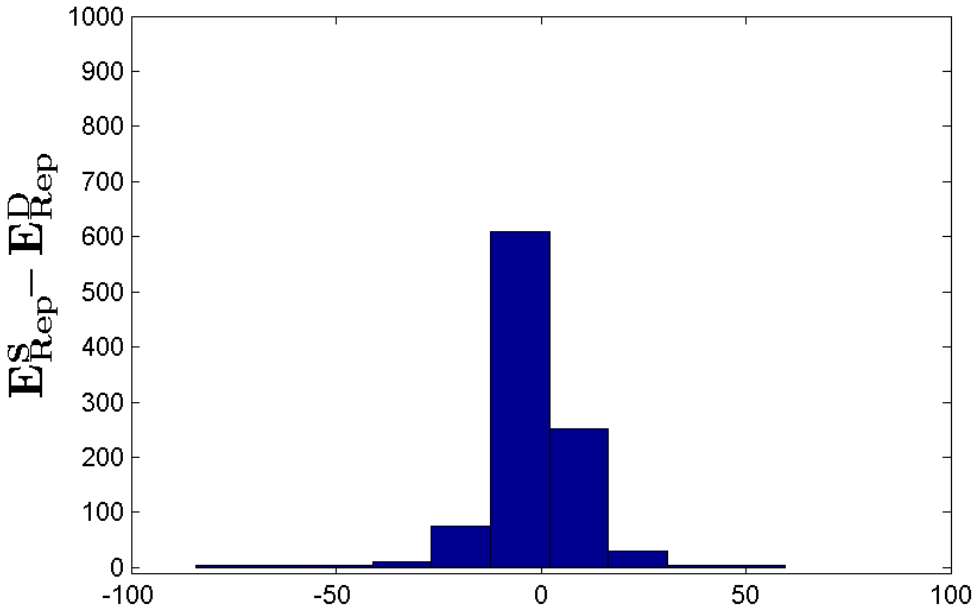


Figure 2.9. SSTATS vs. DUPLEX representative errors.

a DUPLEX and SSTATS training set. However, when there was a difference between the two methods, SSTATS can yield far smaller errors. Therefore, while similar sets are obtained, greater differences in training and test sets were observed from the DUPLEX method further justifying the use of SSTATS.

Overall, the use of a training set selection algorithm reduced all three errors as compared with a randomly selected training set. SSTATS prevailed as the best performing algorithm as it provided more representative training and test sets in the simulations overall; SSTATS statistically outperformed randomly selected training sets and sets created from the CADEX algorithm in all three types of errors. SSTATS also outperformed the DUPLEX algorithm although there was no statistical difference in representative error between the two methods. In the following Section, training and test sets of images were selected in an RPD of the RX detector.

2.5 RX Algorithm Experiment

Reed and Yu [67] developed the RX detector under the assumption that most images display approximately independent and Gaussian characteristics from pixel to pixel. Prior to implementing the RX detector, it is common practice to apply principal components analysis (PCA) to reduce the number of spectra considered. PCA projects the data into a subspace that produces uncorrelated components; the components accounting for the greatest total variance are retained [24]. Next, the RX detector creates a user-defined window around each test pixel considered, x. The mean, μ, and covariance, Σ, of all pixels within the window (excluding x) are used to perform a generalized maximum likelihood ratio test. An RX score is generated for each pixel considered using the following formula:

$$RX\left(x\right) = \left(x-\mu\right)^{T}\left[\left(\frac{N}{N+1}\right)\Sigma + \left(\frac{1}{N+1}\right)\left(x-\mu\right)\left(x-\mu\right)^{T}\right]^{-1}\left(x-\mu\right) \quad (2.48)$$

This process is repeated by selecting a new test pixel and creating a new window to define the background. RX scores are calculated for each test pixel. Since individual pixels are assumed to be independent and Gaussian, these RX scores are compared with $\chi^2_{\alpha,\rho}$ where α is the quantile and ρ is the degrees of freedom of the Chi-squared distribution. The pixels are classified in the following manner.

$$x = \begin{cases} \text{outlier} & \text{if } RX(x) \geq \chi^2_{\alpha,\rho} \\ \text{background} & \text{otherwise} \end{cases} \quad (2.49)$$

2.5.1 Inputs - Control Variables.

The RX detector has three controllable settings which will be varied in a designed experiment to identify robust optimal operating settings. The control factors are described below.

1. Window size (A) – A^2 defines the area of the window surrounding the test pixel used to define the background mean and covariance (an odd number)

2. α (B) – the α parameter selected for the Chi-squared distribution

3. Number of principal components retained (C) – defines the number of principal components kept after PCA

2.5.2 Images - Noise Variables.

Data used for this experiment came from the Hyperspectral Digital Imagery Collection Equipment (HYDICE) sensor Forest Radiance I and Desert Radiance II collection events. Spectral data was collected by the HYDICE sensor in 210 bands encompassing the near-ultraviolet, visible, and infrared spectrums. Due to a small sample size, ten images were halved and used to train and test the RX detector. These image halves were defined by the Fisher ratio, percent targets and number of clusters in the same fashion as Section 2.4.2. The image noise characteristics are broken out with training sets by method in Table 2.3.

2.5.3 Outputs.

There were five potential testable outputs considered for the RX detector: processing time, true positive fraction (TPF), false positive fraction (FPF), label accuracy (LA) and total error (TE). True positive fraction compares the number of correctly identified anomalous pixels with the total number of actual target pixels; false positive fraction compares the total number of falsely labeled pixels (pixels labeled as anomalies when they were actually background) with the total number of background pixels. Label accuracy considers the number of correctly identified anomalous pixels as a percentage of the total number of pixels labeled as anomalous. Total error compares the total number of misclassified pixels to the total number of pixels considered.

Table 2.3. Image noise characteristics.

Image	Half	Fishers Score	Percent Targets	Num Clusters	SSTATS	CAD	DUP	Rand
1D	Upper	1.780	0.004	3		Train	Train	Train
1D	Lower	1.627	0.003	3	Train			Train
1F	Upper	0.433	0.039	6		Train		
1F	Lower	0.315	0.022	10	Train	Train	Train	Train
2D	Upper	0.096	0.025	3		Train		
2D	Lower	0.176	0.029	3	Train		Train	
2F	Upper	0.963	0.008	8	Train		Train	Train
2F	Lower	0.931	0.009	7				
3D	Upper	0.169	0.003	3		Train	Train	Train
3D	Lower	1.430	0.003	3	Train		Train	Train
3F	Upper	0.265	0.005	5	Train			Train
3F	Lower	0.215	0.008	5			Train	
4F	Upper	0.083	0.005	7				
4F	Lower	0.078	0.006	8	Train		Train	Train
4	Upper	1.409	0.016	7		Train		
4	Lower	2.638	0.028	4	Train	Train	Train	
5	Upper	0.266	0.011	6	Train			
5	Lower	1.845	0.005	6		Train	Train	
5F	Upper	0.199	0.008	10		Train		Train
5F	Lower	0.741	0.009	7	Train	Train		Train

Below, all five measures are assessed and reported but interest is centered on maximizing the quotient, $\frac{LA}{TE}$, due to the high FP rate common when applying the RX detector. The ranges for each response are in Table 2.4.

Table 2.4. AutoGAD RPD response ranges.

Output Parameter	Range
TPF	$[0, 1]$
FPF	$[0, 1]$
LA	$[0, 1]$
TE	$[0, 1]$
Time	$[0, \infty]$

While maximizing the objective function, $\frac{LA}{TE}$, to identify robust settings, the objective function is not the primary focus in assessing representative training and test sets. Due to the very nature of the CADEX algorithm, it is expected that there will be a large disparity between the average $\frac{LA}{TE}$ observed in the training and test sets. DUPLEX and SSTATS are expected to display consistent algorithmic performance across their respective training and test sets in terms of representative error. For an example, consider the generic noise displayed in Figure 2.10. The CADEX algorithm is expected to select the most extreme data points for the training set and leave the remaining points for the test set. Whereas, the DUPLEX and SSTATS algorithms are expected to create training and test sets that are more similar.

Figure 2.11 shows the training and test sets selected by the CADEX algorithm. The four extreme points are included in the training set and the test set consists of strictly interior points. Since the training set spans a larger volume of the design space than the test set, it is expected that the training set average performance will be influenced by the extremes not evident in the test set. As such, an average performance on the test set larger than the training set would not be unexpected. Therefore, it appears the CADEX algorithm will not provide representative sets.

Figure 2.12 gives the training and test sets identified by the DUPLEX algorithm.

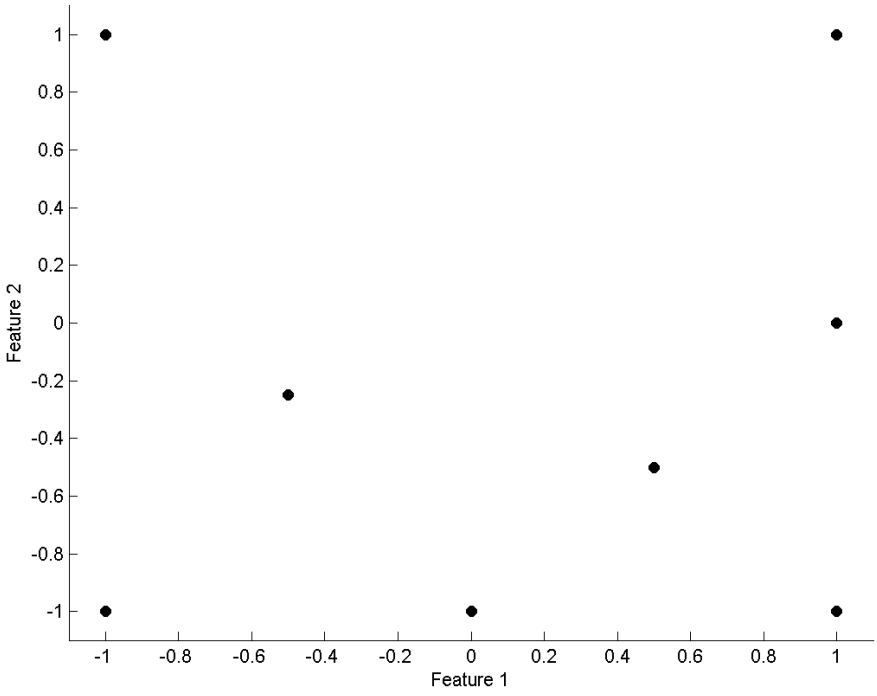


Figure 2.10. Example noise data.

Some extreme points lie in both the training and test sets. Representative error for the DUPLEX algorithm should be smaller than for the CADEX algorithm since both extreme and interior points are distributed roughly equally across both sets.

Finally, Figure 2.13 shows the training and test sets selected by the SSTATS algorithm on the example noise set. SSTATS also includes a mix of interior and extreme points in the training and test sets.

2.5.4 Experimental Design.

There is no variability when using specific settings for RX on a given image. Thus, replications were not required in the experimental design. A full factorial design of the control factors comprised a 5*3*10 run experiment. The ranges tested for each control variable are listed in Table 2.5.

Before applying any regression methods, the control variables were all transformed

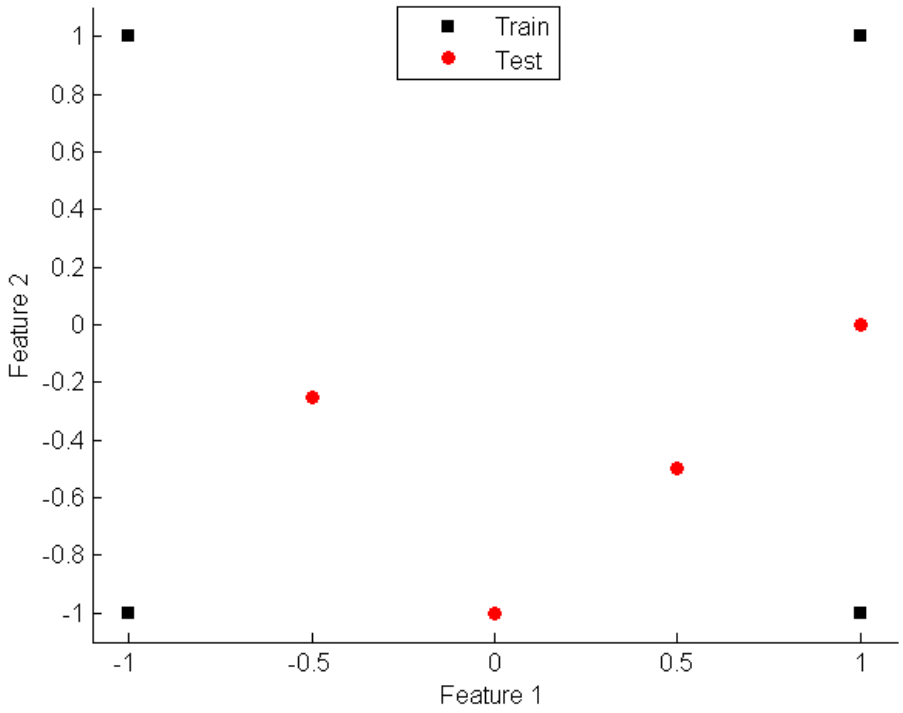


Figure 2.11. CADEX training and test sets for example noise.

Table 2.5. RX RPD response ranges.

Input Parameter	Type	Test Range	Factor Levels
Window size (A)	Discrete	$[17, 25]$	3
α (B)	Continuous	$[1 \times 10^{-10}, 1 \times 10^{-1}]$	10
Number of PCs retained (C)	Discrete	$[8, 12]$	5

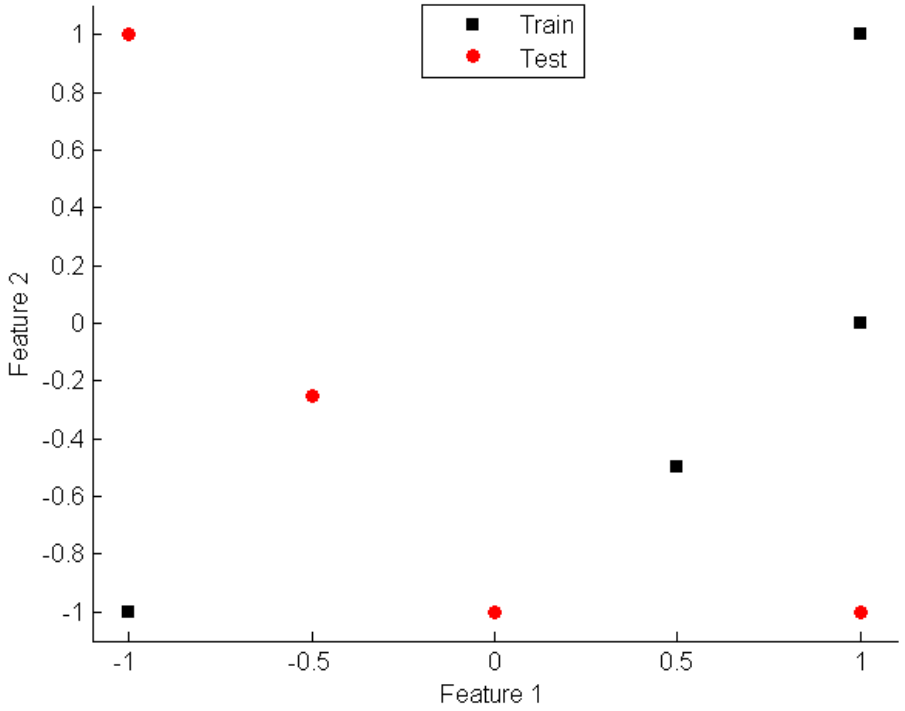


Figure 2.12. DUPLEX training and test sets for example noise.

to coded variables in [-1,1]. This step was performed using

$$x_{i,j} = \frac{\xi_{i,j} - [\max(\xi_{i,j}) + \min(\xi_{i,j})]/2}{[\max(\xi_{i,j}) - \min(\xi_{i,j})]/2} \tag{2.50}$$

where $x_{i,j}$ is exemplar i of the coded noise variable j and $\xi_{i,j}$ is the original value [63].

2.5.5 Results.

The RPD coefficient estimates based upon all four training set selection techniques and the response of interest, $\frac{LA}{TE}$, are in Table 2.6. With the exception of some of the coefficients found using a random training set, most coefficient estimates were consistent across the four techniques.

The optimal control settings for the four models are given in Table 2.7. The random model had markedly different settings than the other methods. Based on the simulation experiment results in Figures 2.6-2.9, the SSTATS and DUPLEX models

Table 2.6. RX RPD coefficient estimates.

	SSTATS	CADEX	DUPLEX	Random
β_0	46.2	28.0	45.7	45.6
β_1	2.4	4.7	3.3	4.8
β_2	-15.8	-9.4	-15.5	-16.1
β_3	-1.9	-1.2	-1.3	-2.5
γ_1	-20.3	0	-9.7	-18.3
γ_2	-13.8	-13.8	-14.3	-30.8
γ_3	4.2	3.1	14.2	26.7
δ_{11}	0	0	3.9	4.0
δ_{21}	6.6	0	3.0	7.4
δ_{31}	0	0	0	0
δ_{12}	-2.4	-4.4	-3.6	-3.3
δ_{22}	6.0	5.5	6.2	10.3
δ_{32}	0	0	0	0
δ_{13}	0	1.9	3.1	6.9
δ_{23}	-2.3	-1.9	-5.5	-8.2
δ_{33}	0	0	0	0
B_{11}	-4.1	0	-4.8	-4.0
B_{12}	0	-0.9	0	0
B_{13}	0	0	0	0
B_{21}	0	0	0	0
B_{22}	0	0	0	0
B_{33}	0	0	0	0
RMSE	47.6	33.8	48.3	46.1

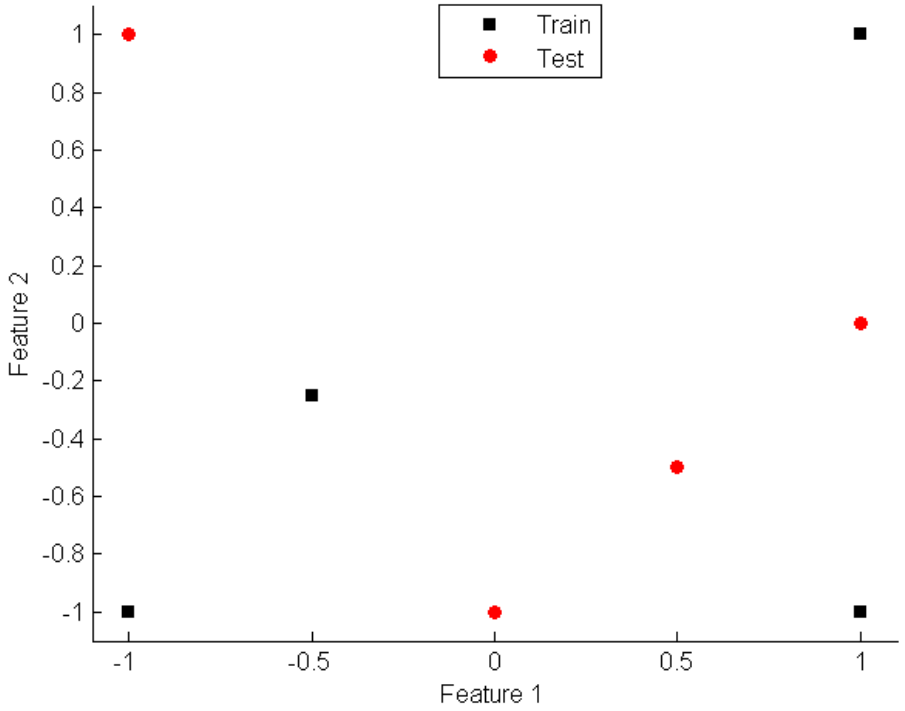


Figure 2.13. SSTATS training and test sets for example noise.

were expected to provide the most representative training and test sets (smallest representative error, E^t_{Rep}) with the SSTATS model providing the best overall fit for the model (smallest location and fit errors, E^t_{Loc} and E^t_{Fit}). In the RX application, the SSTATS representative error was the smallest in comparison with the other models. The random training set representative error was small, but its overall LT values for the random training and test were far larger than any other method. An additional estimate of the similarity between training and test sets can be computed by considering the ratio of the determinants of the $X'X$ matrices for both sets [79, pg. 421]. A ratio of $\frac{|X'_{tr}X_{tr}|}{|X'_{te}X_{te}|} = 1$ implies the two sets span equal volume in noise variable space. The SSTATS algorithm yielded a ratio near one while the other algorithms had much larger ratios. Table 2.7 arrays the model fits and LT values observed on the RX algorithm.

Table 2.7. RX results.

	SSTATS	**DUPLEX**	**CADEX**	**Random**
Window Size	23	21	25	17
α	1×10^{-10}	1×10^{-10}	1×10^{-10}	0.1
Number of PCs	8	8	9	8
Train LT	57041	62273	69461	89345
Test LT	56580	55210	39317	88502
Representative Error	461	7063	30144	843
$\lvert X'_{tr}X_{tr}\rvert / \lvert X'_{te}X_{te}\rvert$	1.0087	3.3516	10.0288	6.2818

Individual comparisons of method performance broken out by training and test sets are shown in Tables 2.8-2.11. In these tables, five responses are reported, the emphasis here is on $\frac{LA}{TE}$. Overall, the individual results closely matched the results presented in Table 2.7. Table 2.8 gives the results from using the SSTATS method. The SSTATS response of interest, $\frac{LA}{TE}$, was consistent between the training and test sets as shown by the representative error in Table 2.7. There was considerable variability from image to image as was expected.

Table 2.9 gives the individual image results from training with images selected using the CADEX algorithm. The algorithm creates a very diverse training set leaving the rest of the images in the test set. As expected, the training and test sets were not representative of one another as is shown in Table 2.7.

Table 2.10 gives the individual image results from the DUPLEX algorithm. This method was expected to compete closely with SSTATS in terms of representative error as reflected in Table 2.7. However, the SSTATS algorithm was able to produce somewhat more representative sets than the DUPLEX algorithm.

Table 2.11 shows the individual image results from the randomly selected training set. The results emphasize the importance of using a training set selection strategy rather than just using a random draw. The randomly selected set produced the worst overall responses for $\frac{LA}{TE}$ averaging around 2.0 (the other methods averaged over 70).

Table 2.8. SSTATS image results.

	Image	Image Half	TPF	FPF	LA	TE	LA/TE
Train	1D	Upper	0.02	0.0013	0.07	0.01	13.35
	1F	Upper	0	0.0001	0	0.04	0
	2D	Upper	0.16	0.0002	0.95	0.02	45.21
	2F	Upper	0.18	0.0002	0.87	0.01	122.09
	3D	Lower	0.02	0.0015	0.26	0.03	7.71
	3F	Upper	0.24	0.0001	0.90	0.00	218.72
	3F	Lower	0.22	0.0005	0.67	0.00	174.62
	4	Lower	0.00	0.0005	0.18	0.04	5.01
	5	Upper	0.06	0.0003	0.67	0.01	62.78
	5F	Upper	0.11	0.0004	0.65	0.01	88.46
Train Avg			0.10	0.0005	0.52	0.02	**73.80**
Test	1D	Lower	0.02	0.0013	0.05	0.01	10.32
	1F	Lower	0.02	0.0003	0.64	0.03	22.46
	2D	Lower	0.17	0.0002	0.96	0.02	42.84
	2F	Lower	0.24	0.0006	0.64	0.00	146.45
	3D	Upper	0.22	0.0022	0.26	0.00	53.74
	4F	Upper	0.24	0	1	0.00	286.47
	4F	Lower	0.25	0	1	0.01	148.89
	4	Upper	0	0.0010	0	0.02	0
	5	Lower	0.05	0.0004	0.56	0.01	49.26
	5F	Lower	0.06	0.0005	0.73	0.02	34.15
Test Avg			0.13	0.0007	0.58	0.01	**79.46**

Table 2.9. CADEX image results.

	Image	Image Half	TPF	FPF	LA	TE	LA/TE
Train	1D	Upper	0.04	0.0013	0.12	0.01	22.14
	1F	Upper	0	0.0002	0	0.04	0
	1F	Lower	0.02	0.0004	0.57	0.03	19.87
	2D	Upper	0.18	0.0002	0.96	0.02	46.55
	3D	Upper	0.27	0.0017	0.34	0.00	81.71
	4	Upper	0	0.0010	0	0.02	0
	4	Lower	0.00	0.0004	0.3	0.04	8.31
	5	Lower	0.07	0.0003	0.7	0.01	63.49
	5F	Upper	0.16	0.0004	0.77	0.01	111.56
	5F	Lower	0.08	0.0005	0.78	0.02	37.04
Train Avg			0.08	0.0007	0.45	0.02	**39.07**
Test	1D	Lower	0.02	0.0015	0.05	0.01	8.56
	2D	Lower	0.19	0.0002	0.96	0.02	44.44
	2F	Upper	0.27	0.0002	0.93	0.01	147.08
	2F	Lower	0.29	0.0006	0.70	0.00	171.73
	3D	Lower	0.02	0.0016	0.24	0.03	7.06
	3F	Upper	0.26	0.0001	0.91	0.00	227.96
	3F	Lower	0.23	0.0006	0.63	0.00	160.93
	4F	Upper	0.22	0.0001	0.89	0.00	237.66
	4F	Lower	0.25	0.0001	0.95	0.01	138.49
	5	Upper	0.06	0.0003	0.68	0.01	64.27
Test Avg			0.18	0.0005	0.69	0.01	**120.82**

Table 2.10. DUPLEX image results.

	Image	Image Half	TPF	FPF	LA	TE	LA/TE
Train	1D	Upper	0.02	0.0015	0.05	0.01	8.04
	1F	Lower	0.02	0.0003	0.62	0.03	21.45
	2D	Lower	0.12	0.0004	0.89	0.02	37.33
	2F	Upper	0.16	0.0002	0.86	0.01	118.43
	3D	Upper	0.20	0.0022	0.23	0.00	46.18
	3D	Lower	0.01	0.0016	0.21	0.03	6.11
	3F	Lower	0.22	0.0005	0.67	0.00	174.62
	4F	Lower	0.22	0	1	0.01	143.57
	4	Lower	0	0.0007	0	0.04	0
	5	Lower	0.02	0.0005	0.35	0.01	29.88
Train Avg			0.10	0.0008	0.49	0.02	**58.56**
Test	1D	Lower	0.01	0.0013	0.03	0.01	5.26
	1F	Upper	0	0.0001	0	0.04	0
	2D	Upper	0.13	0.0004	0.89	0.02	40.87
	2F	Lower	0.21	0.0006	0.63	0.00	141.11
	3F	Upper	0.2	0.0001	0.89	0.00	205.12
	4F	Upper	0.19	0	1	0.00	267.37
	4	Upper	0.01	0.0008	0.13	0.02	8.19
	5	Upper	0.04	0.0003	0.6	0.01	55.70
	5F	Upper	0.08	0.0004	0.63	0.01	83.68
	5F	Lower	0.04	0.0004	0.67	0.02	30.41
Test Avg			0.09	0.0004	0.55	0.01	**83.77**

Table 2.11. Random training set image results.

	Image	Image Half	TPF	FPF	LA	TE	LA/TE
Train	1D	Upper	0.71	0.0640	0.05	0.06	0.70
	1D	Lower	0.79	0.0762	0.04	0.08	0.51
	1F	Lower	0.20	0.0558	0.09	0.08	1.22
	2F	Upper	0.75	0.0562	0.10	0.06	1.74
	3D	Upper	0.34	0.0716	0.02	0.07	0.22
	3D	Lower	0.22	0.0588	0.11	0.08	1.35
	3F	Upper	0.88	0.0516	0.08	0.05	1.58
	4F	Lower	0.44	0.0408	0.09	0.05	1.97
	5F	Upper	0.60	0.0578	0.08	0.06	1.25
	5F	Lower	0.20	0.0583	0.07	0.07	0.99
Train Avg			0.51	0.0591	0.07	0.07	**1.09**
Test	1F	Upper	0.11	0.0573	0.07	0.09	0.79
	2D	Upper	0.84	0.0264	0.44	0.03	14.94
	2D	Lower	0.77	0.0297	0.42	0.03	11.88
	2F	Lower	0.88	0.0561	0.07	0.06	1.27
	3F	Lower	0.68	0.0582	0.05	0.06	0.80
	4F	Upper	0.70	0.0410	0.07	0.04	1.75
	4	Upper	0.18	0.0568	0.05	0.07	0.69
	4	Lower	0.19	0.0399	0.15	0.07	2.24
	5	Upper	0.28	0.0594	0.05	0.07	0.75
	5	Lower	0.6	0.0593	0.11	0.06	1.66
Test Avg			0.52	0.0484	0.15	0.06	**2.55**

This experiment serves as an illustration that SSTATS performs well in a small sample size problem utilizing non-orthogonal data. The CADEX algorithm yielded excellent test set results, but clearly there was a difference between the training and test sets as shown in the disparate representative error. The DUPLEX algorithm created training and test sets with improved representative error in comparison with the CADEX algorithm, but the SSTATS algorithm produced the most similar training and test sets. The SSTATS algorithm had the lowest average test $\frac{LA}{TE}$ due to the fact that the algorithm included representative extreme points in the training and test sets. The randomly selected training set had a lower representative error than CADEX and

DUPLEX, but the LT scores from the random set were extremely large indicating the potential for poor future performance when randomly selecting a training set.

2.6 Conclusions

Selecting training and test sets of hyperspectral images for use in RPD of anomaly detection algorithms is a very complex problem, especially when limited data is available. Previous research considered each image as a categorical variable and identified optimized settings based on each training image. This chapter used discrete and continuous image noise characteristics to more adequately define training and test sets of images. The hyperspectral image noise features were not orthogonal requiring a new method of identifying well separated sets of images. An objective function was constructed to find the most representative training and test sets with small sample size for model validation. The space of all possible training and test sets was searched. Both simulation and RX results produced reduced errors by applying the SSTATS method rather than using the CADEX or DUPLEX algorithms or randomly selecting training and test sets of images.

III. Optimizing Hyperspectral Imagery Anomaly Detection Algorithms through Improved Robust Parameter Design Considering Noise by Noise Interactions

3.1 Introduction

Hyperspectral sensors provide data rich environments essential to solve numerous problems arising in such areas as military applications, oceanography, forestry, urban planning, and cartography [49]. The analysis of hyperspectral data often follows a sequence of time-intensive processes between acquisition by the sensor to final analysis [47]. For example, an unmanned aerial vehicle can be used in real-time to identify panchromatic image chips containing anomalous pixels presumably containing man-made objects [82]. The image chips provide a cue for an analyst to match specific materials based on their reflectance spectra in the image chip with a list of objects in a library. With the myriad of possible spectra associated with the object library as well as the intricacies involved in atmospheric compensation [81], the task of analyzing large amounts of image chips can be daunting. Therefore, accurate anomaly detection algorithms which identify pixels with spectrally distinct signatures as compared with surrounding pixels, are of paramount importance as the percentage of image chips containing true anomalous objects of interest occur with low probabilities [14]. Inaccurate anomaly detection algorithms produce image chips of background objects for analysis which tie up valuable resources. Further, anomaly detector performance varies due to numerous factors including altitude and scene background. Thus, the need arises to identify robust anomaly detector settings capable of yielding consistent responses across varied image backgrounds. Landgrebe [48] summarized this concept for future hyperspectral algorithms:

> ...what is needed is an analysis process that is robust in the sense that it would work effectively for data of a wide variety of scenes and conditions,

> and can be used effectively by users rather than only by producers of the technology. The algorithms do not need to be simple, but they must be simple to apply and robust against user problems [48, pg. 419].

Anomaly detectors are relatively simple to implement as they require no *a priori* signature information and typically fall into two categories depending on the estimate of the background. Local models define the background based on a local neighborhood around a test pixel while global models typically specify a background distribution from across the entire image, or a large section of the image [81]. Some examples of local background models are the Reed-Xiaoli (RX) detector [67], the locally adaptive iterative RX detector [84] and the support vector data description (SVDD) [7]. Some global background models include the gaussian mixture model generalized likelihood ratio test (GMM-GLRT) [81], orthogonal subspace projection RX [13] and the autonomous global anomaly detector (AutoGAD) [37]. Regardless of the anomaly detector being utilized, algorithm performance is often negatively impacted by uncontrollable noise factors which introduce additional variance into the process. The noise variables are considered uncontrollable in real-world applications, but assumed as able to be fixed for a designed experiment. In the case of hyperspectral imagery (HSI), the noise variables are embedded in the image under consideration. For instance, two images of the same scene taken at different times of day will have different sun angle effects introducing variability into the spectral data collected [47]. Landgrebe [46] defined noise in remote sensing systems by the atmospheric effect, sensor detector/preamplifier noise processes and quantization noise. Mindrup *et al.* [57] developed a framework of continuous and discrete noise characteristics to describe images based on three measurable noise characteristics: the Fisher score, the percent of target pixels and the number of clusters. These characteristics, used in this chapter, were then used to select training and test sets of images [58].

Most, if not all, anomaly detection algorithms require a user to identify some initial parameters. These parameters (or controls) affect the overall algorithm performance. In general, anomaly detector performance can be viewed as a function of controllable and uncontrollable factors plus random process noise, ϵ, that yields a response indicating anomaly or background. Equation 3.1 shows this relationship with x and z defined as controllable and uncontrollable factors respectively.

$$y = F(x, z) + \epsilon \tag{3.1}$$

The model in Equation 3.1 fits directly into the robust parameter design (RPD) framework. RPD seeks to choose controllable parameter settings that produce responses that are not sensitive to changes attributed to noise variables [63]. Typically RPD models assume that no quadratic noise ($z_i z_i$) or noise by noise interactions ($z_i z_j$ for $i \neq j$) exist. This chapter will refer to both as noise by noise ($N \times N$) interactions. The RPD model for $N \times N$ interactions is developed in this chapter and a practice example is provided where $N \times N$ terms are necessary.

Anomaly detection algorithm effectiveness is judged based on summary statistics of the algorithm performance. Some of the more common summary statistics used are classification accuracy and label accuracy. True positive fraction (TPF) is a typical measure employed from an engineering, or designer, viewpoint to a system while label accuracy (LA) reflects a user viewpoint [29]. TPF is strictly concerned with how many pixels in the image are correctly labeled. LA assesses how many pixels labeled as anomalous are actually anomalies. In practice, TPF must be balanced by the number of false positives produced by the model. In an extreme case, one may obtain perfect TPF by changing the classification threshold to identify everything as anomalies. Obviously, the anomaly detector would be useless since an analyst would then have to examine every pixel within an image. In general, TPF

is improved by allowing more indications of anomalous pixels while LA is improved by reducing the number of anomaly indications (keeping only anomaly indications with high confidence) thereby helping to ensure pixels labeled as anomalous are truly anomalies. Thus, when considering LA, the total number of regions of interest (image chips) identified might be reduced, but confidence in the pixels labeled as anomalies is greatly improved. This chapter considers both viewpoints with a response variable incorporating both LA and TPF.

This chapter is organized as follows. Section 3.2 reviews RPD concepts and develops the $N \times N$ extension. Section 3.3 describes AutoGAD and the RPD experiment performed. Results from using the standard RPD model as well as the new model including $N \times N$ are provided. Finally, Section 3.4 concludes the chapter.

3.2 Robust Parameter Design

RPD methods were developed to identify robust process control settings capable of consistent performance in the presence of uncontrollable or noise factors. It is assumed that noise factors are uncontrollable in practice, but can be controlled in designed RPD experiments [63]. The overall true process model can be described as a function of control variables, x, and noise variables, z.

$$y = G(x, z) \tag{3.2}$$

Lin and Tu [51] proposed a criterion considering the process mean and variance as an estimate for mean square error (MSE) to solve for optimal control variable settings in RPD problems. The Lin and Tu MSE minimization criterion (LT) considers the process mean with respect to a target value, T, and process variance, as shown below.

$$LT_{z,\epsilon}\left(G(x,\cdot)|x\right) = \left\{E_{z,\epsilon}\left(G(x,\cdot)|x\right) - T\right\}^2 + var_{z,\epsilon}\left(G(x,\cdot)|x\right) \tag{3.3}$$

The vector of optimal control variable settings, x^*, can be identified by solving

the following constrained optimization problem

$$x^* = \arg\min_{x \in \mathbf{D}} \quad LT_{z,\epsilon}\left(G(x,\cdot)|x\right) \tag{3.4}$$

where the vector of control variables, x, is constrained to the experimental design space, $\mathbf{D}$, which is a closed and bounded compact set.

In the rest of this section, a standard RPD model, $y^{(1)}$, is presented. Then, an extension to RPD considering $N \times N$ interactions, $y^{(2)}$, is developed and supporting examples are provided.

3.2.1 Standard RSM Model ($y^{(1)}$).

Typically, second-order models are developed in response surface methodology approaches to RPD and higher order control interactions are ignored due to the sparsity of effects principle [63]. Noise by noise interactions and squared noise terms are also assumed to be negligible. A general matrix form of the quadratic response surface model proposed by Myers [63] to approximate $G_1(x,z)$ is

$$y^{(1)} = G_1(x,z) = \beta_0 + x'\beta + x'Bx + z'\gamma + x'\Delta z + \epsilon \tag{3.5}$$

where $y^{(1)}$ represents the standard RPD model, x is an $r_{\mathbf{x}} \times 1$ vector of control variables, z is an $r_{\mathbf{z}} \times 1$ vector of noise variables, β_0 is the intercept, β is an $r_{\mathbf{x}} \times 1$ vector of control variable coefficients, B is an $r_{\mathbf{x}} \times r_{\mathbf{x}}$ matrix of the quadratic control coefficients, γ is an $r_{\mathbf{z}} \times 1$ vector of noise variable coefficients, Δ is an $r_{\mathbf{x}} \times r_{\mathbf{z}}$ matrix of control by noise interaction coefficients and ϵ is a random error assumed to be normally distributed $N(0, \sigma^2 I_{r_{\mathbf{z}}})$; $r_{\mathbf{x}}$ and $r_{\mathbf{z}}$ represent the number of control and noise factors respectively. The noise variables, $z = (z_1, z_2, \ldots, z_{r_z})$, are assumed to be a vector of independent random variables with $E(z_i) = 0 \quad \forall \quad i$ and $var(z) = \sigma_z^2 I_{r_z}$ which is easily accomplished by centering and scaling. The expected value model,

with respect to z, for the estimated quadratic model in Equation (3.5) becomes

$$E(y^{(1)}|x) = E_{z,\epsilon}\left(G_1(x,\cdot)|x\right) = \beta_0 + x'\beta + x'Bx. \tag{3.6}$$

For the remainder of this section, $E(y^{(1)}|x)$ represents short-hand notation for $E_{z,\epsilon}\left(G_1(x,\cdot)|x\right)$. Similarly, the variance model of Equation (3.5) is given by

$$\begin{aligned} var(y^{(1)}|x) &= var_{z,\epsilon}\left(G_1(x,\cdot)|x\right) \\ &= \left(\gamma' + x'\Delta\right) var_z(z) \left(\gamma' + x'\Delta\right)' + \sigma^2 \\ &= \sigma_z^2 \left(\gamma' + x'\Delta\right)\left(\gamma' + x'\Delta\right)' + \sigma^2. \end{aligned} \tag{3.7}$$

For the remainder of this section, $var(y^{(1)}|x)$ represents short-hand notation for $var_{z,\epsilon}\left(G_1(x,\cdot)|x\right)$. The corresponding LT criterion becomes

$$\begin{aligned} LT(y^{(1)}|x) &= LT_{z,\epsilon}\left(G_1(x,\cdot)|x\right) \\ &= \left(\beta_0 + x'\beta + x'Bx - T\right)^2 + {\sigma_z}^2 \left(\gamma' + x'\Delta\right)\left(\gamma' + x'\Delta\right)' + \sigma^2. \end{aligned} \tag{3.8}$$

For the remainder of this section, $LT(^{(1)}|x)$ represents short-hand notation for $LT_{z,\epsilon}\left(G_1(x,\cdot)|x\right)$. The noise parameters, z, effect the overall LT criterion in the variance model through the noise parameter coefficients, γ and Δ, but the criterion is completely in terms of control parameters, x. Thus, optimal control settings can be identified through constrained optimization as in Equation 3.4 [41, 69]. An extended RPD model including $N \times N$ interactions is now introduced.

3.2.2 RPD Model Including $N \times N$ ($y^{(2)}$).

If we allow for the assumption that $cov(z_i, z_j) \neq 0 \quad$ for some $i \neq j$ (implying $cov(z) = \Sigma_z$) and expand the response surface model to include both squared noise terms ($z_i z_i$) and noise by noise interaction terms ($z_i z_j$ for $i \neq j$), the new general matrix form of the quadratic response surface model including $N \times N$ to approximate

$G_2(x, z)$ is

$$y^{(2)} = G_2(x, z) = \beta_0 + x'\beta + x'Bx + z'\gamma + x'\Delta z + z'\Phi z + \epsilon \tag{3.9}$$

where $y^{(2)}$ represents the extended RPD model and Φ is a matrix of the $N \times N$ coefficients and the rest of the terms are as described previously in Equation (3.5). The expected value model, with respect to z, for the estimated quadratic model in Equation (3.9) is

$$E\left(y^{(2)}|x\right) = \beta_0 + x'\beta + x'Bx + E[z'\Phi z] \tag{3.10}$$

Searle [72] showed when $x \sim N(0, V)$, $E[x'Ax] = tr(AV)$ where tr signifies the trace of a matrix. Since the noise variables are assumed to be distributed $z \sim N(0, \Sigma_z)$, the expected value model becomes

$$E\left(y^{(2)}|x\right) = \beta_0 + x'\beta + x'Bx + tr(\Phi\Sigma_z). \tag{3.11}$$

The variance model for $Y^{(2)}$ can be written as

$$\begin{aligned} var\left(y^{(2)}|x\right) &= (\gamma' + x'\Delta)\Sigma_z(\gamma' + x'\Delta)' + \sigma^2 + var(z'\Phi z) \\ &+ 2cov((\gamma' + x'\Delta)z, z'\Phi z). \end{aligned} \tag{3.12}$$

The variance model in Equation (3.12) is the same as the variance model in Equation (3.7) with the addition of two terms, $2cov((\gamma' + x'\Delta)z, z'\Phi z)$ and $var(z'\Phi z)$. The

term, $2cov((\gamma' + x'\Delta)z, z'\Phi z)$, can be rewritten by letting $\alpha' = \gamma' + x'\Delta$ to be

$$\begin{aligned} 2cov\left(\alpha' z, z'\Phi z\right) &= 2\left(E(\alpha' z z'\Phi z) - E(\alpha' z)E(z'\Phi z)\right) \\ &= 2E(\alpha' z z'\Phi z) \\ &= 2E\left(\sum_{i=1}^{m}\sum_{j=1}^{m}\sum_{k=1}^{m}\alpha_k z_k z_j \phi_{ji} z_i\right) \\ &= 2\sum_{i=1}^{m}\sum_{j=1}^{m}\sum_{k=1}^{m}\alpha_k \phi_{ji} E(z_k z_j z_i). \end{aligned} \tag{3.13}$$

This results in three types of terms multiplied by a constant. These terms are $E(z_a^3)$, $E(z_a^2 z_b)$ and $E(z_a z_b z_c)$ respectively. Anderson showed that all three types of resulting terms are zero because with multivariate normal data, "any third moment about the mean is zero" [2]. Therefore, the entire added covariance term is zero.

Searle [72] also showed when $x \sim N(0, V)$, $var(x'Ax) = 2tr(AVAV)$. Therefore, the term $var(z'\Phi z)$

$$var(z'\Phi z) = 2tr\left(\Phi\Sigma_z\Phi\Sigma_z\right) \tag{3.14}$$

Thus, the variance model for $Y^{(2)}$ with $N \times N$ interactions becomes

$$var\left(y^{(2)}|x\right) = (\gamma' + x'\Delta)\Sigma_z(\gamma' + x'\Delta)' + 2tr\left(\Phi\Sigma_z\Phi\Sigma_z\right) + \sigma^2. \tag{3.15}$$

Finally, the LT criterion for the $N \times N$ model becomes

$$\begin{aligned} LT(y^{(2)}|x) &= \left(\beta_0 + x'\beta + x'Bx + tr(\Phi\Sigma_z) - T\right)^2 \\ &+ (\gamma' + x'\Delta)\Sigma_z(\gamma' + x'\Delta)' + 2tr\left(\Phi\Sigma_z\Phi\Sigma_z\right) + \sigma^2. \end{aligned} \tag{3.16}$$

3.2.3 Example.

A simple example is presented to show the impact of significant $N \times N$ interactions if ignored. The fitted model of Myers and Montgomery [63, pg. 577] was selected as the true function to be approximated with some additional $N \times N$ coefficients: ϕ_{11}, ϕ_{22} and ϕ_{12}. However, only 2 noise variables were used and the z_3 terms which appeared

in the original problem were deleted. The assumed true response relationship was taken as

$$\begin{aligned} y &= 30.382 - 2.925x_1 - 4.136x_2 + 2.855x_1x_2 + 2.596x_1^2 + 2.715x_2^2 \\ &+ 2.736z_1 - 2.326z_2 - 0.278x_1z_1 + 0.893x_1z_2 + 1.999x_2z_1 \\ &+ 1.430x_2z_2 + \phi_{11}z_1^2 + \phi_{22}z_2^2 + \phi_{12}z_1z_2. \end{aligned} \tag{3.17}$$

Initially all of the $N \times N$ coefficients (ϕ_{11}, ϕ_{22} and ϕ_{12}) were set to zero meaning $y^{(1)}$ from Equation (3.5) and $y^{(2)}$ in Equation (3.9) are equivalent. $N \times N$ interactive effects were added to this model incrementally in the following fashion:

$$\Phi_{(n)} = \Phi_{(n-1)} + \begin{bmatrix} 0.25 & -0.125 \\ -0.125 & 0.25 \end{bmatrix}; \Phi_{(0)} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \tag{3.18}$$

Two replicates of a 3^2 factorial design for the control variables were crossed with ten non-orthogonal noise variables producing the complete experimental design. A residual error of $\sigma^2 = 2$ was used to generate training and test data. Parameter coefficients were fit for both RPD models. Model performance was assessed based on R^2 and absolute fit error. Absolute fit error is defined as

$$E_f = |LT(y|x^*) - LT(y|x_t^*)| \tag{3.19}$$

where the parameters in the LT model are from the true function parameters in Equation (3.17), x^* is the vector of true optimal control settings and x_t^* is the vector of estimated optimal control settings from either the $y^{(1)}$ or $y^{(2)}$ model. Figure 3.1 gives an example of fit error. The true LT surface is plotted across levels for a single control variable. The true optimal point, x^*, and estimated optimal point, x_t^*, are labeled on the x-axis. The fit error is the difference in the true LT surface evaluated at x^* and x_t^*.

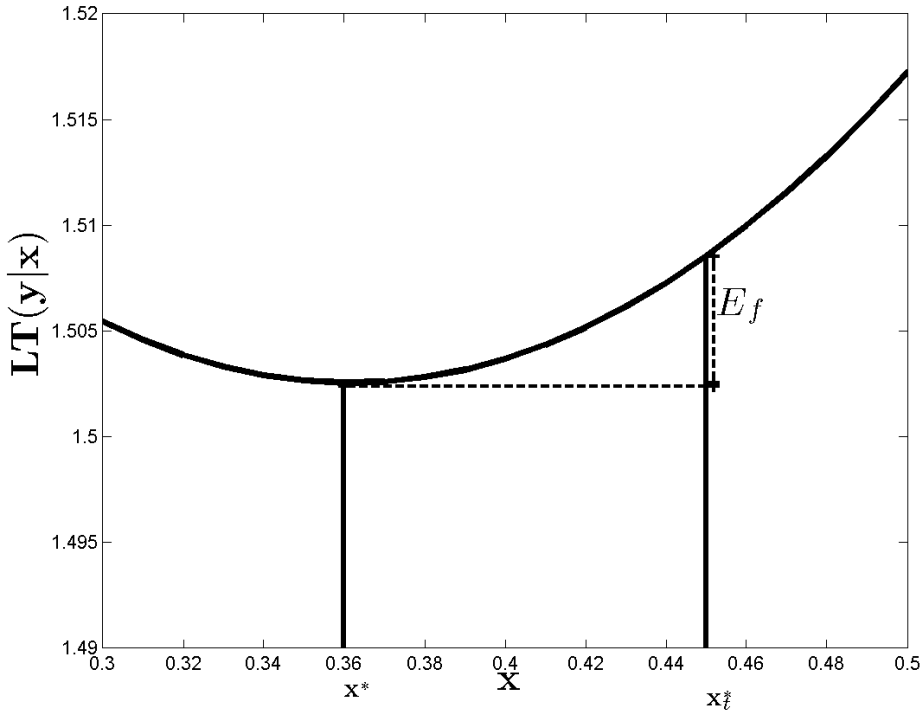


Figure 3.1. Example of fit error.

Figure 3.2 compares R^2 for the $y^{(1)}$ and $y^{(2)}$ models. The plot shows the change in R^2 as $N \times N$ effects are increased. The R^2 for the $y^{(1)}$ model drops to near 0.5 after 30 increments of increasing $N \times N$ effects; conversely, the R^2 for the $y^{(2)}$ model remains high as $N \times N$ effects are increased.

Figure 3.3 displays the Euclidean distance between the estimated optimal settings and the true optimal settings for both RPD models, defined as location error. The $y^{(1)}$ model location error is consistent with the $y^{(2)}$ model location error until $N \times N$ effects become significant (around increment 11).

Figure 3.4 displays the fit errors for the two models. When $N \times N$ is not a large factor, there is no significant difference between the two models in terms of fit error. Once the $N \times N$ effects become significant, around increment 11, the $y^{(1)}$ model is no longer able to approximate the surface appropriately; the estimated optimal point

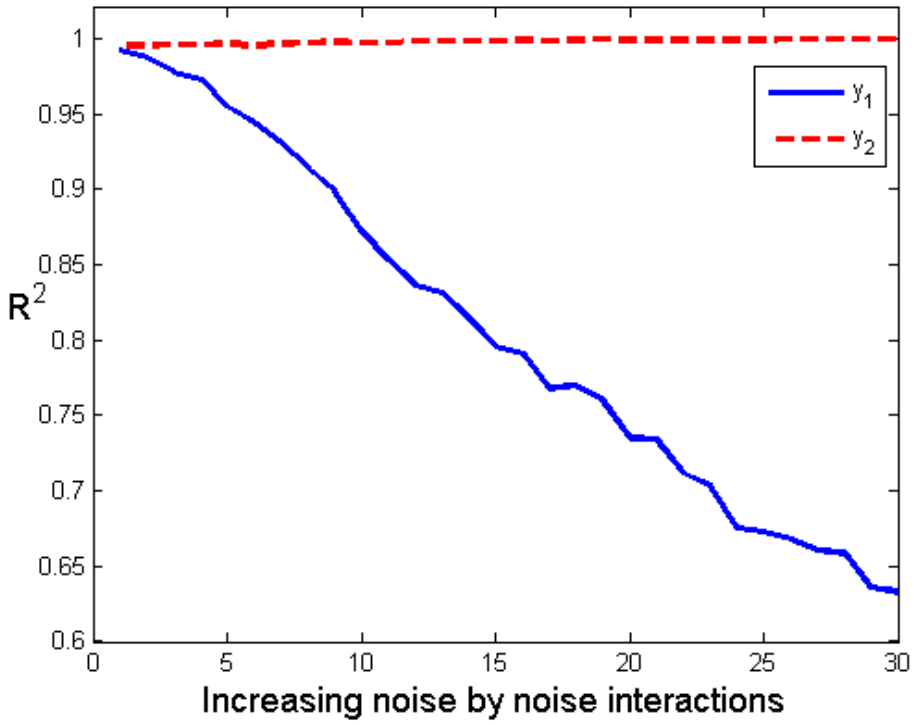


Figure 3.2. Effect of increased $N \times N$ on R^2.

from the $y^{(1)}$ model is always moved to an extreme of the design space and the LT value using the true model parameters at the estimated optimum from $y^{(1)}$ becomes extremely large.

3.2.4 Computer Network Performance Example.

Another example from Schmidt and Launsby [71] presented in Myers *et al.* [63, prob. 6.8] is embellished to demonstrate the perils of ignoring $N \times N$ interactions in RPD modeling. The original data is included in Appendix A: Table 1.1. This problem is only intended to display the potential for finding differing robust control settings due to the RPD model selected. The problem considers performance data from an integrated circuit/packet-switched computer network using response surface techniques. Four design variables were considered in the experiment: circuit switch arrival rate (CS), packet switch arrival rate (PS), voice call service rate (Serv) and

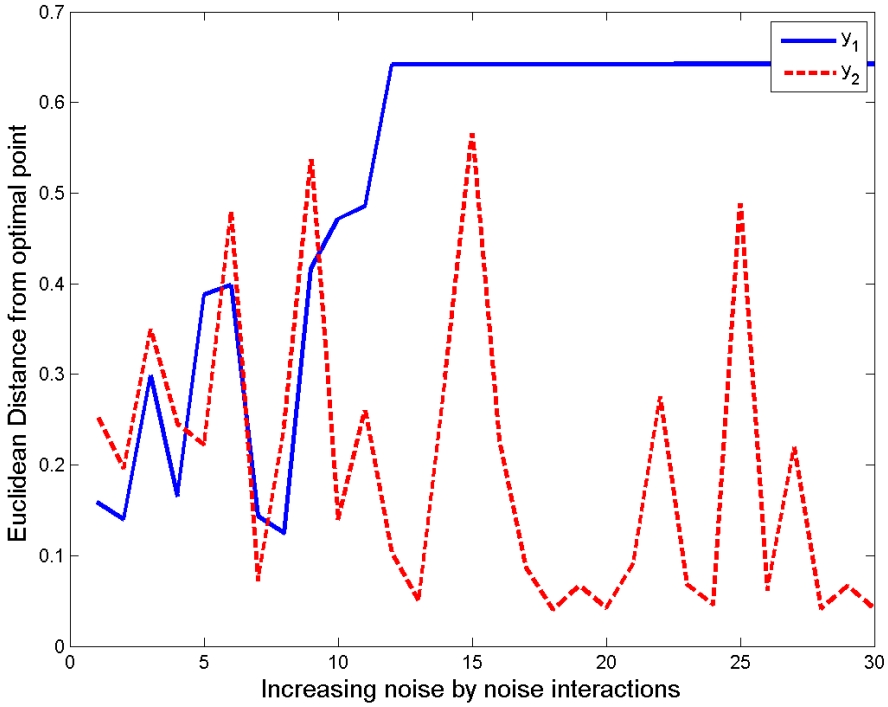


Figure 3.3. Effect of increased $N \times N$ on optimal settings.

the number of slots per link (Slots). Two responses were recorded, but the focus is strictly on the fraction of voice calls blocked (BLK).

Suppose the circuit switch arrival rate and voice call service rate are treated as noise variables. RPD models for $y^{(1)}$ and $y^{(2)}$ were generated for the BLK response. However, since BLK is a proportion, it was first transformed by [3]

$$BLK' = arcsin(\sqrt{BLK}). \tag{3.20}$$

There was a significant difference between the $y^{(1)}$ and $y^{(2)}$ models as shown in Table 3.1. The $y^{(2)}$ model which includes $N \times N$ interactions explained 14% more variance in R^2 and reduced root mean square error (RMSE). The optimal number of slots was 46 for both models while the packet switch arrival rate varied.

The actual coefficients for each model are arrayed in Table 3.2. Adding $N \times N$ in

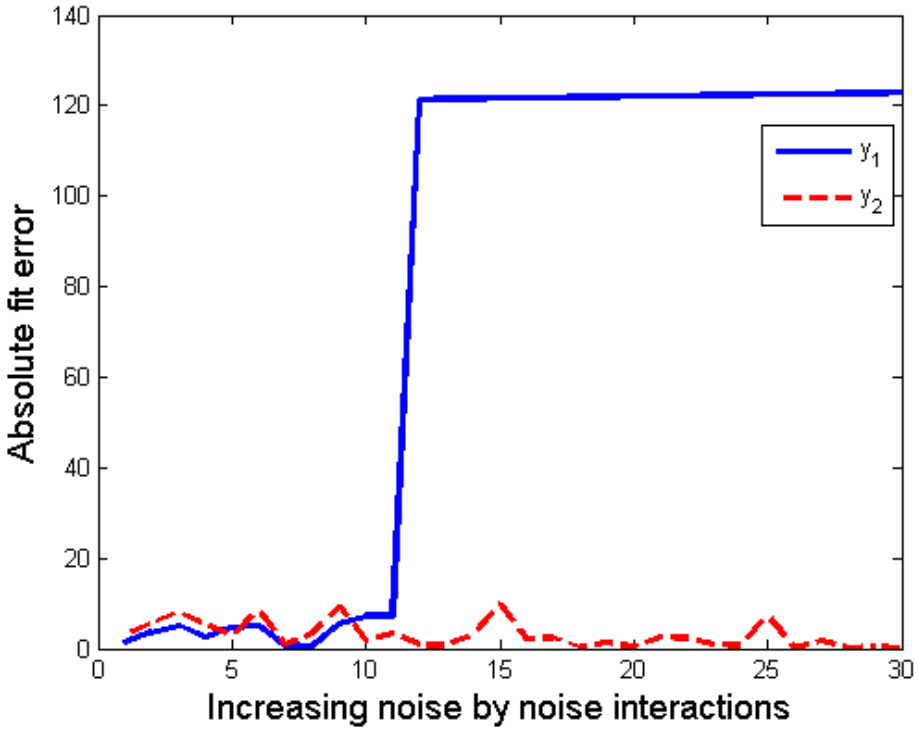


Figure 3.4. Effect of increased $N \times N$ on fit error.

Table 3.1. Computer network model fits.

	$\mathbf{y^{(1)}}$	$\mathbf{y^{(2)}}$
$\mathbf{R^2}$	0.7942	0.9081
Adj $\mathbf{R^2}$	0.7599	0.883
RMSE	0.0599	0.0419
PS	150	289
Slots	46	46

the $y^{(2)}$ model increased the number of significant terms by three.

Table 3.2. Computer network model coefficients.

	$\mathbf{y^{(1)}}$	$\mathbf{y^{(2)}}$
β_0	0.1189	0.1409
β_1	0.1291	0.0644
β_2	-0.0689	-0.0568
γ_1	0.0741	0.1309
γ_2	0	0.076
δ_{11}	0.0867	0.0329
δ_{21}	-0.0433	-0.0315
δ_{12}	0.0501	0
δ_{22}	0	0
B_{11}	0	0.2284
B_{12}	-0.0206	-0.016
B_{22}	0	0
Φ_{11}	N/A	-0.1524
Φ_{12}	N/A	0.0328
Φ_{22}	N/A	0

Overall, the LT surfaces for both models displayed closely matched expected value models. Figure 3.5 displays the LT surface for the $y^{(1)}$ model and Figure 3.6 provides the LT surface for the $y^{(2)}$ model. Optimal settings for the $y^{(1)}$ and $y^{(2)}$ models are denoted in both figures by a circle and diamond respectively. As was shown in this example, considerably different optimal control settings may be identified based on the RPD model selected.

3.3 Autonomous Global Anomaly Detector

AutoGAD is designed to isolate pixels which are spectrally different from the background pixels. The algorithm is based on the global linear mixture model [54]. AutoGAD employs techniques to automate feature extraction, feature selection and target pixel identification. There are several user selected parameters within the algorithm which are detailed in section 3.3.6. In the rest of this section, the four phases of the AutoGAD algorithm are reviewed following the description in Johnson

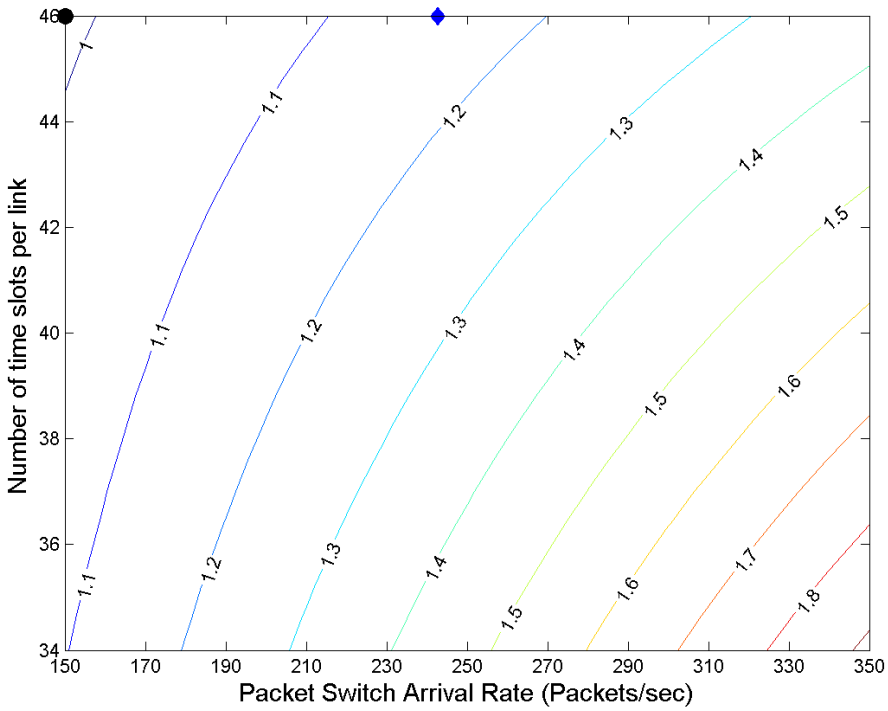


Figure 3.5. LT surface plot for $y^{(1)}$ model.

[37]. This is followed by control and noise variable definitions and AutoGAD outputs are described for RPD. Next, the experimental design is presented. Finally, AutoGAD performance using parameters selected from the $y^{(1)}$ and $y^{(2)}$ RPD models are compared.

3.3.1 Image Preprocessing.

A hyperspectral image, also called an image cube, consists of p spectral bands of an $m \times n$ spatial pixel representation of a sensed area. Each pixel in the spectral dimension represents an intensity of energy reflected back to the sensor. All spectral dimensions for a given pixel represent a potential target signature. This cube is first reshaped from a three-dimensional image into an $m \times n$ row and p column matrix of feature vectors. Next absorption bands are removed reducing the dimensionality from p to g spectra. For the images considered, specific absorption bands have been

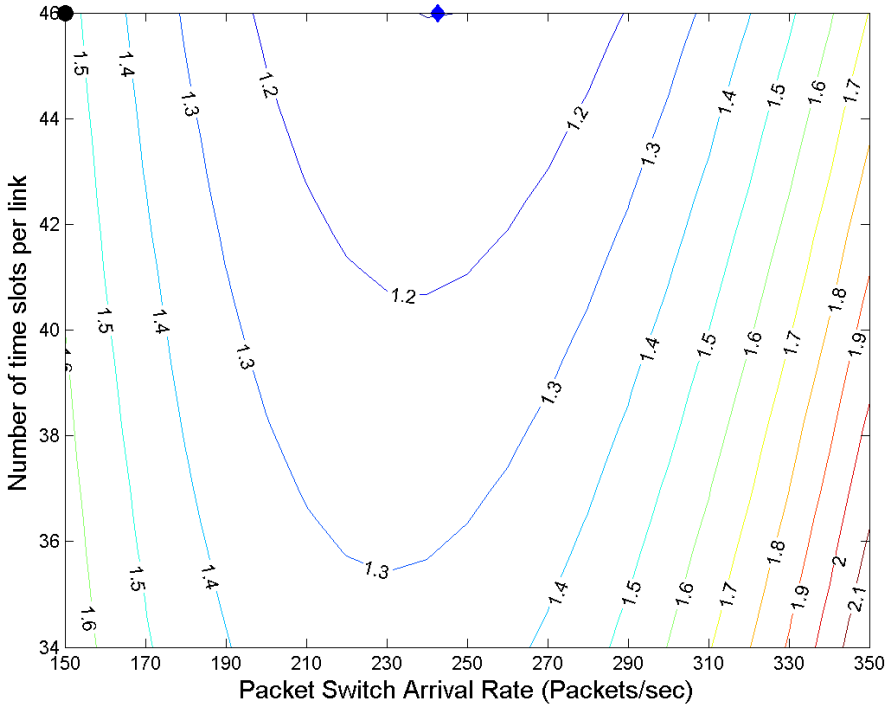


Figure 3.6. LT surface plot for $y^{(2)}$ model.

specified by Smetek [75]. The result of the preprocessing step is a matrix representation of the image cube with a subset of total spectra included. This process is shown pictorially in Figure 3.7. Here, the initial image cube contained 210 spectral bands with only 145 remaining after absorption bands were removed [37].

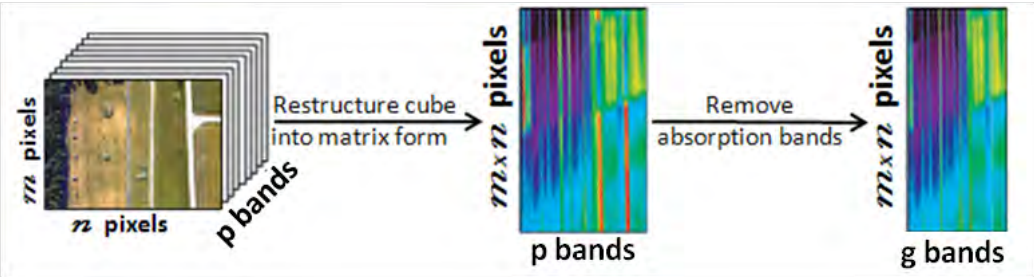


Figure 3.7. AutoGAD preprocessing [56].

3.3.2 Step 1: Feature Extraction I.

After the absorption bands have been removed, the dimensionality is further reduced by utilizing Principal Components Analysis (PCA). PCA projects the data into a subspace that produces uncorrelated components; the components accounting for the greatest total variance are kept by the algorithm [24]. Previous anomaly detection algorithms selected the number of bands to keep based on a user defined threshold of accountable variance. Johnson [37] demonstrated the variability explained by the number of spectral dimensions kept by using a single variance threshold was not adequate for anomaly detection. Instead, his algorithm identifies the required number of spectral bands using a Maximum Distance Secant Line (MDSL) algorithm. This algorithm identifies the "knee in the curve" of a plot of ordered eigenvalues. Next, the data is whitened implying that the data is centered at 0 and scaled with unit variance. The process is depicted in Figure 3.8. The total number of dimensions is reduced from g to k (typically less than 15) [37].

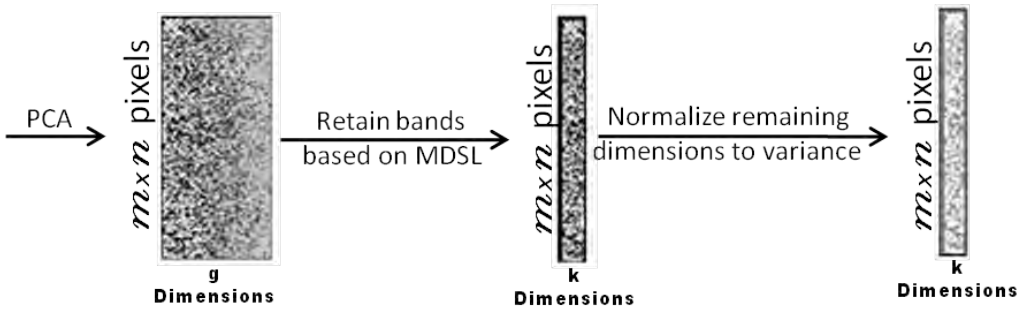


Figure 3.8. AutoGAD PCA [56].

3.3.3 Step 2: Feature Extraction II.

Next AutoGAD performs independent component analysis (ICA), a linear mixture model which results in vectors that are independent [36, 83]. These independent vectors signify specific endmembers composing the image. Abundance maps for each

endmember are created by reshaping each independent vector back to an $m \times n \times 1$ pixel image. The intended result of both feature extraction steps is a set of independent and uncorrelated components with some components representing a combination of specific discernable spectra in the image and others capturing noise which can be filtered to reduce the feature set further. Thresholds are then specified to identify which abundance maps have the highest potential in flagging anomalies. Figure 3.9 depicts the result of ICA [37].

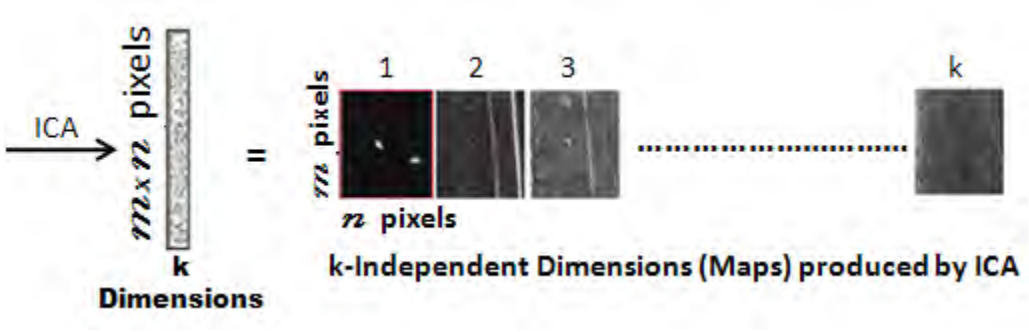


Figure 3.9. AutoGAD ICA [56].

3.3.4 Step 3: Feature Selection.

Figure 3.9 makes it obvious to the eye that the feature vectors in map 1 spotlight true outliers while the other abundance maps highlight noise and other non anomalous features. AutoGAD employs a clever way to allow a computer program select the abundance maps that are believed to contain true anomalies. Two thresholds are defined. The first is the maximum pixel intensity observed for a feature vector. A histogram is constructed of all pixel intensities. Chiang [18] found that anomaly pixels typically had intensities greater than the first empty histogram bin. The second is the potential anomaly signal to noise ratio (PA SNR). A noise floor is derived from a histogram of pixel intensities within the specified component. Background pixels should have values close to zero, their mean; anomalies should be sparse and create

a long skinny tail in components containing more than just noise. The first bin to the right of zero with no pixel intensities present is selected as the noise floor. The resulting potential anomaly signal to noise ratio is calculated as

$$PA\ SNR = -10 \log \frac{\text{var (potential anomaly signal)}}{\text{var (noise)}}. \tag{3.21}$$

Johnson found that components exceeding both thresholds were most likely to contain true anomaly pixels. These components (or abundance maps when the vector is reshaped back into an m row $\times$ n column $\times$ 1 independent uncorrelated vector) were kept for further processing. Figure 3.10 depicts an example of the feature selection step; in this example, the feature selection step resulted in a dimensionality reduction from nine to four potential anomaly maps [37]. The first spectral band displayed in Figure 3.10 is selected as a potential anomaly map because the maximum pixel intensity and PA SNR are both above their selected thresholds. The second spectral band is identified as noise. The maximum intensity is much lower with a shorter tail and the PA SNR is negative both indicating a non-anomaly map.

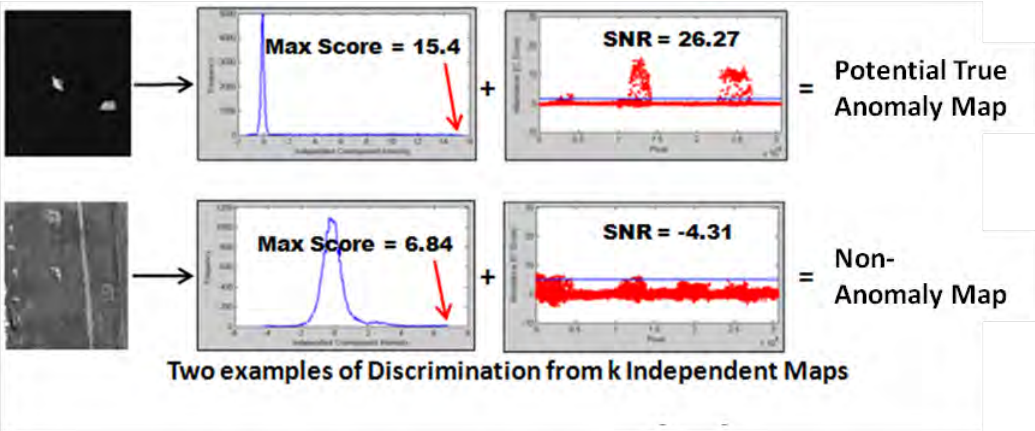


Figure 3.10. AutoGAD feature selection [56].

3.3.5 Step 4: Identification.

Johnson improves classification results further by applying an adaptive Wiener filter [50] to smooth out the background noise. The AutoGAD algorithm iteratively utilizes an adaptive Wiener noise filter to compare each pixel value and the variance of all pixels in a window to the variance across the entire image. Pixels with large variance with respect to the rest of the image maintain large intensities while pixels with small variance are assumed to be noise and are smoothed out (multiplied by a fraction to reduce their pixel intensity). This process continues for a prespecified number of iterations producing a final set of independent uncorrelated components. The noise floor is set again based on the first zero bin in each histogram. All pixels with intensities larger than this noise floor are considered as anomalies. An example of this process is shown in Figure 3.11. In the rightmost part of Figure 3.11, white in the combined map represents pixels correctly labeled as anomalies (TP) and gray represents pixels misclassified as anomalies (FP) [37].

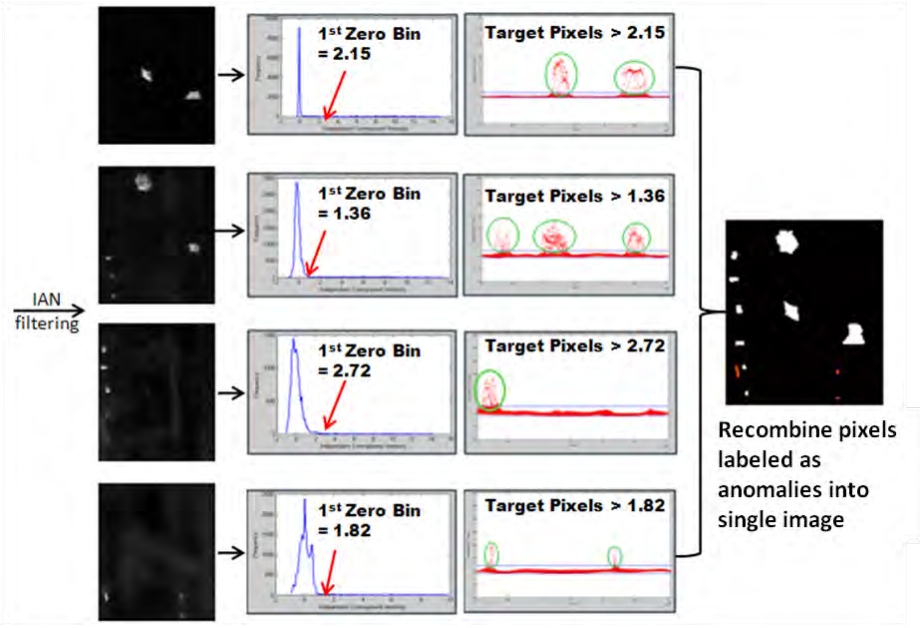


Figure 3.11. AutoGAD target pixel ID [56].

3.3.6 Inputs - Control Variables.

AutoGAD has nine controllable settings, five of which will be varied in a designed experiment to identify optimal operating settings. The control factors are described below [37]:

1. Dimension adjust (A)–increases/decreases the number of dimensions kept from MDSL

2. Max score threshold (B)–threshold from feature selection step for identifying maps containing potential anomalies

3. Bin width SNR (C)–defines the histogram bin width used to create a SNR in feature selection step

4. PA SNR threshold (D)–potential anomaly SNR threshold used in feature selection to identify potential anomaly maps

5. Bin width identify (E)–defines histogram bin width used to define the noise floor in identification phase

6. Smooth iterations high (F)–number of iterations for IAN filtering when PA SNR is above a threshold

7. Smooth iterations low (G)–number of iterations for IAN filtering when PA SNR is below a threshold

8. Low SNR (H)–threshold to decide whether smooth iterations high or low is used

9. Window size (J)–defines the size of the neighborhood applied in the IAN process of the identification phase

3.3.7 Images - Noise Variables.

Data used for this experiment came from the Hyperspectral Digital Imagery Collection Equipment (HYDICE) sensor Forest Radiance I and Desert Radiance II collection events. Spectral data was collected by the HYDICE sensor in 210 bands encompassing the near-ultraviolet, visible, and infrared spectrums. Due to a small sample size, ten images were halved and used to train and test AutoGAD from the dataset. These image halves were defined by three observable noise characteristics identified by Mindrup *et al.* [57]: Fisher ratio, ratio of targets and number of clusters.

The Fisher ratio, $\mathbf{z}_1$, was described by Duda *et al.* [25, 55] is a measure for the discriminating power of a variable. The Fisher ratio for an individual image, $i = 1, 2, \ldots, I$ where I is the total number of images under consideration, is defined as the average Fisher ratio across each image band, $k = 1, 2, \ldots, K$. Thus, the Fisher ratio for image i is

$$z_{i1} = \frac{\sum_{k=1}^{K} \left(\frac{\left(\mu_{a_{i,k}} - \mu_{b_{i,k}}\right)^2}{\sigma^2_{a_{i,k}} + \sigma^2_{b_{i,k}}} \right)}{K} \tag{3.22}$$

where $\mu_{a_{i,k}}$ and $\sigma^2_{a_{i,k}}$ are the mean and variance of the anomalous pixels, a, in band k of image i and $\mu_{b_{i,k}}$ and $\sigma^2_{b_{i,k}}$ are the mean and variance of the background pixels, b, in band k of image i, all defined from a truth mask.

The ratio of anomalous pixels, $\mathbf{z}_2$, was calculated if there was a truth map for each image, $i = 1, 2, \ldots, I$, by

$$z_{i2} = \frac{v_i}{b_i} \tag{3.23}$$

where v_i and b_i represent the number of anomalous pixels and background pixels in image i, respectively.

The number of clusters represents the number of homogenous groups of pixels within an image. The number of clusters, $\mathbf{z}_3$, was recorded for each image, $i = 1, 2, \ldots, I$ using the X-means algorithm as developed by Pelleg and Moore [66].

Each noise feature vector was standardized by

$$\hat{\mathbf{z}}_k = \frac{\mathbf{z}_k - \mu_{\mathbf{z}_k}}{\sigma_{\mathbf{z}_k}} \tag{3.24}$$

where $\mu_{\mathbf{z}_k}$ and $\sigma_{\mathbf{z}_k}$ represent the mean and standard deviation of the k^{th} noise vector, $\mathbf{z}_k$. The three standardized noise feature vectors were combined in an $I \times q$ noise matrix, $Z = [\hat{\mathbf{z}}_1 \quad \hat{\mathbf{z}}_2 \quad \hat{\mathbf{z}}_3]$, with I total images and $q = 3$ noise variables.

Typical experimental designs employ orthogonal designs implying the design can be bounded by a p-dimensional hypercube. [39] Hyperspectral imagery noise variables are different from the standard noise variables used in RPD due to the fact that the variables cannot be controlled for a designed experiment. Images must be chosen such that the training and test sets are representative of one another. Thus, the training set selection methodology described in Mindrup *et al.* [58] was utilized. The image noise characteristics are broken out by training and test set in Table 3.3. Two additional images free of anomalies were considered as a separate validation set. These additional validation images were expected to provide a better assessment of true algorithm performance due to the assumption that most images will contain few if any actual anomalies of interest. The validation images were also halved and also summarized in Table 3.3.

3.3.8 Outputs.

In addition to the nine control variables, there are five relevant outputs from AutoGAD: processing time, true positive fraction (TPF), false positive fraction (FPF), label accuracy (LA) and the total number of correct clusters of anomalies detected. True positive fraction compares the number of correctly identified anomalous pixels with the total number of actual target pixels; false positive fraction compares the total number of falsely labeled (labeled as anomalies when they were actually background) pixels with the total number of background pixels. Label accuracy considers

Table 3.3. Image noise characteristics.

	Image	Image half	Fisher ratio	Percent targets	Number of clusters
Training set	1D	Top	1.7797	0.0043	3
	1F	Top	0.4335	0.0392	5
	2D	Top	0.0957	0.0247	4
	2F	Top	0.9633	0.0084	7
	3D	Bottom	1.4299	0.0033	3
	3F	Top	0.265	0.0053	8
	3F	Bottom	0.2153	0.0078	5
	4	Bottom	2.6382	0.0275	4
	5	Top	0.2658	0.0109	6
	5F	Top	0.1991	0.0078	10
Test set	1D	Bottom	1.6265	0.0028	3
	1F	Bottom	0.3148	0.0225	5
	2D	Bottom	0.1762	0.0288	3
	2F	Bottom	0.9311	0.0085	7
	3D	Top	0.1695	0.0034	3
	4F	Top	0.0826	0.0046	7
	4F	Bottom	0.0779	0.0063	8
	4	Top	1.4093	0.0156	6
	5	Bottom	1.8451	0.0052	4
	5F	Bottom	0.7412	0.0094	7
Validation set	1C	Top	NaN	0	10
	1C	Bottom	N/A	0	10
	2C	Top	N/A	0	9
	2C	Bottom	N/A	0	9

the number of correctly identified anomalous pixels as a percentage of the total number of pixels labeled as anomalous. Clusters of pixels identified as anomalies were also considered. If at least one pixel of a cluster labeled as an anomaly fell within a true anomaly cluster, the anomalous cluster was considered to have been identified. If none of the pixels within a cluster contained a true anomalous pixel, the entire cluster was considered a FP as it would force an analyst to review an image chip without any objects of interest. Occasionally a particular AutoGAD setting run against an image would not identify any pixels as anomalies and the label accuracy for these instances was taken as zero.

All five measures were examined but a combination of label accuracy and true positive fraction was employed in an effort to consider both the engineering and user points of view shown below in Equation (3.25).

$$y = LA + TPF. \tag{3.25}$$

The ranges for each response are in table 3.4.

Table 3.4. AutoGAD RPD response ranges.

Output Parameter	Range
TPF	$[0, 1]$
FPF	$[0, 1]$
LA	$[0, 1]$
Time	$[0, \infty]$
Num correct clusters	$[0, \infty]$

3.3.9 Experimental Design.

Due to the large number of variables, a screening design was used to identify the primary factors of interest and thus, the total number of experimental runs required. The preliminary results showed four factors that could be fixed at a single setting yielding the best overall response across a wide array of images: Dimension Adjust,

both Smooth parameters and Window Size. Three of the four fixed variable settings matched those suggested by Johnson [37]. This left all five continuous control variables for further study. The ranges used for each control variable are displayed in Table 3.5. Each control factor was varied across three equally spaced levels.

Table 3.5. AutoGAD RPD factor ranges.

Input Parameter	Type	Classification	Test Range
Dimension Adjust (A)	Discrete	Fixed	-2
Max Score Threshold (B)	Continuous	Control	[6,14]
Bin Width SNR (C)	Continuous	Control	[0.01,0.09]
PA SNR Threshold (D)	Continuous	Control	[6 14]
Bin Width Identify (E)	Continuous	Control	[0.01,0.09]
Smooth Iterations High (F)	Discrete	Fixed	100
Smooth Iterations Low (G)	Discrete	Fixed	20
Low SNR (H)	Continuous	Control	[6,14]
Window Size (J)	Discrete	Fixed	3

Before applying any regression methods, the control variables were all transformed to coded variables in [-1,1]. This step was performed using

$$x_{i,j} = \frac{\xi_{i,j} - [\max(\xi_{i,j}) + \min(\xi_{i,j})]/2}{[\max(\xi_{i,j}) - \min(\xi_{i,j})]/2} \tag{3.26}$$

where $x_{i,j}$ is exemplar i of the coded noise variable j and $\xi_{i,j}$ is the original value [63].

A face centered cube (FCC) design was selected for the control variables allowing estimation of quadratic effects with five center runs; an example of the FCC for two control variables in coded variables is in Table 3.6. The FCC was then crossed with ten training images and replicated ten times for a total of 4250 experimental runs. Replications were necessary because the ICA algorithm applies a random component to AutoGAD. Borror *et. al.* [10] showed that while an FCC is not the most economical experimental design, it is comparable in terms of prediction error to other D-optimal and G-optimal designs considered for statistical designs with noise variables. The design considered here is slightly modified from that in Borror *et. al.* due to the

crossed nature of the noise variables with the control FCC.

Table 3.6. Example FCC for two control variables.

x_1	x_2
-1	-1
1	-1
-1	1
1	1
-1	0
1	0
0	-1
0	1
0	0
0	0
0	0
0	0
0	0

Multicollinearity issues arose when $N \times N$ was introduced to the model. A heuristic was applied to remove columns with high variance inflation factors (VIF). First, the VIF for all columns in the $y^{(2)}$ model was calculated. If at least one VIF was greater than ten, the column with the largest VIF was deleted prior to fitting a regression model. In case of a tie, higher order terms were removed first. This process was repeated until all VIF values were less than ten. For this problem, three columns were deleted due to high VIF scores: Fisher ratio×Fisher ratio (K*K), Fisher ratio×number of clusters (K*M) and percent targets×number of clusters (L*M). The three remaining potential $N \times N$ terms were added to the regression model. Stepwise regression was then used to fit the $y^{(1)}$ and $y^{(2)}$ models from equations (3.5) and (3.9) respectively.

3.3.10 Results.

Table 3.7 gives the overall model fits for the $y^{(1)}$ and $y^{(2)}$ models as well as the respective parameter estimates. Coefficients that were insignificant in both models

were not included for clarity. The overall R^2 increased from 0.47 in the $y^{(1)}$ model to 0.63 in the $y^{(2)}$ model with a similar change in the adjusted R^2 values. Root mean square error was also reduced from 0.36 to 0.30 by adding the $N \times N$ terms. Including $N \times N$ in the model added five terms: bin width for identification×Low SNR (E*H), max×max (B*B), Fisher ratio×percent of targets (K*L), percent of targets×percent of targets (L*L) and number of clusters×number of clusters (M*M). The coefficients for noise terms varied from one model to the other. This was due to the fact that the noise terms were not orthogonal to each other.

Both models were significant with p-values less than 0.0001. However, both models displayed significant lack of fit due to nonconstant variance. This can be seen in the residual versus predicted plot for $y^{(1)}$ in Figure 3.12 ($y^{(2)}$ had a similar plot). The nonconstant variance was an artifact of the bounded response variables. Although the regression model assumption of constant variance was invalid, the models were still useful in selecting optimal AutoGAD settings.

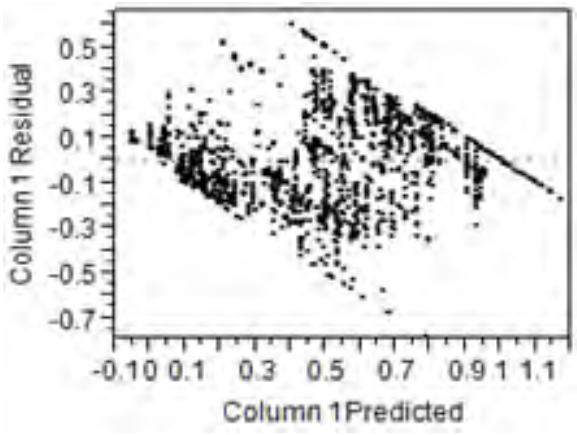


Figure 3.12. AutoGAD $y^{(1)}$ residual versus predicted plot.

Optimal settings for the appropriate model, $y^{(1)}$ or $y^{(2)}$, were calculated using Equation (3.4). The optimal settings for both models as well as the settings suggested by Johnson [37] are in Table 3.8. In general, the optimal settings for the

Table 3.7. AutoGAD fits and coefficient estimates.

Term	$y^{(1)}$ **model**	$y^{(2)}$ **model**
R^2	0.47	0.63
Adjusted R^2	0.47	0.63
Root Mean Square Error	0.36	0.30
Intercept	1.146	1.443
Max (B)	-0.062	-0.062
Bin Width (C)	0.012	0.012
PA SNR (D)	0.013	0.013
Bin Width ID (E)	0.077	0.077
Low SNR (H)	0.026	0.026
Fisher Ratio (K)	-0.039	-0.015
Percent Targets (L)	0.288	0.543
Number of Clusters (M)	-0.025	0.011
B*C	-0.017	-0.017
B*E	0.048	0.048
C*D	0.015	0.015
C*E	-0.014	-0.014
E*H		0.010
B*K	-0.043	-0.043
B*L	0.053	0.053
B*M	0.108	0.108
C*L	-0.030	-0.030
E*K	-0.077	-0.077
E*L	-0.013	-0.013
H*K	0.013	0.013
H*L	-0.013	-0.013
H*M	0.022	0.022
B*B		-0.033
E*E	-0.056	
K*L		-0.023
L*L		-0.257
M*M		-0.058

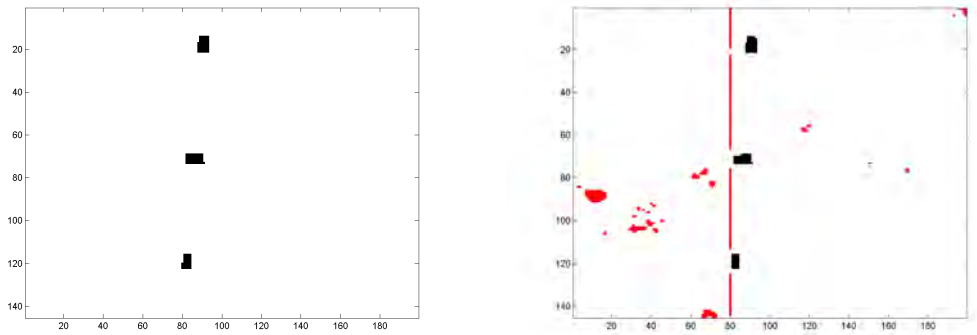
$y^{(1)}$ and $y^{(2)}$ models varied across all five continuous control variables considered. Johnson's settings, selected after extensive experience with the AutoGAD algorithm, were chosen for a higher true positive rate while still considering the other responses. Johnson developed the suggested AutoGAD settings based on algorithm performance observed on the entire set of images and thus has an advantage over those selected using either RPD model since both RPD models only trained on half of the images. Therefore, a direct comparison between the results from either the $y^{(1)}$ or $y^{(2)}$ model and Johnson's settings is not possible. Results on image halves from Johnson's settings are only provided to show the potential change observed when applying an RPD model.

Table 3.8. AutoGAD optimal settings.

	Johnson	$y^{(1)}$	$y^{(2)}$
Dim (A)	0	-2	-2
Max (B)	10	10.9013	8.7696
Bin Width (C)	0.05	0.09	0.0633
PA SNR (D)	2	1	6
Bin Width ID (E)	0.05	0.0248	0.0685
Smooth High (F)	100	100	100
Smooth Low (G)	20	20	20
Low SNR (H)	10	10.0933	14
Window (J)	3	3	3

Tables 2.1, 2.2 and 2.3 (Appendix B) provide detailed results for each image. On average, results from the $y^{(1)}$ model closely mirrored those of Johnson's optimal settings. Both had an overall average TPF of 0.68 with LAs of 0.44 and 0.46 respectively. The averages for the $y^{(2)}$ model were 0.67 for TPF and 0.61 for LA. Thus, the settings identified by including $N \times N$ in the $y^{(2)}$ model improved label accuracy by roughly 15% while only losing 1% in true positive fraction. The performance difference is spotlighted by comparing the results graphically from all three settings on individual images. "Truth masks" for each image were created at the Air Force Institute of

Technology by zooming in on each image and "truthing" individual pixels based on best guesses. Figure 3.13 depicts the "truth mask" for upper half of the 1D image (training set) as well as the performance from all three settings. Pixels shaded black in the figures represent true positives while pixels shaded red represent false positive indications. The $y^{(2)}$ model's increased label accuracy is reflected by the significant reduction in false positives in comparison to the Johnson and $y^{(1)}$ models.



(a) Truth mask

(b) Johnson results

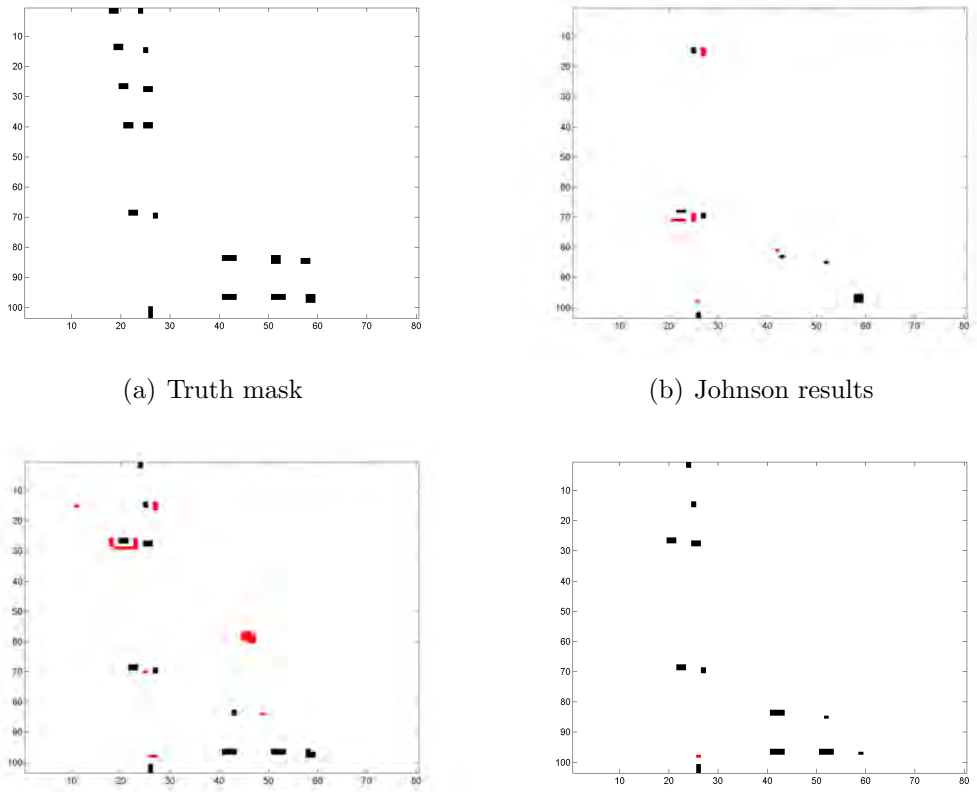
(c) $y^{(1)}$ model results

(d) $y^{(2)}$ model results

Figure 3.13. ARES1D upper half AutoGAD results.

Figure 3.14 depicts similar results from the lower half of image 4F. This image was in the test set and reveals the importance of the $y^{(2)}$ model by including $N \times N$ interactions. A simple RPD model, $y^{(1)}$, is capable of identifying most of the actual anomalies in the image, but at a cost of high false positives. The settings from Johnson again yield similar results to the $y^{(1)}$ model. The results from the $y^{(2)}$ model

show a large reduction in false positives while improving the total number of true anomalous clusters correctly identified.



(a) Truth mask

(b) Johnson results

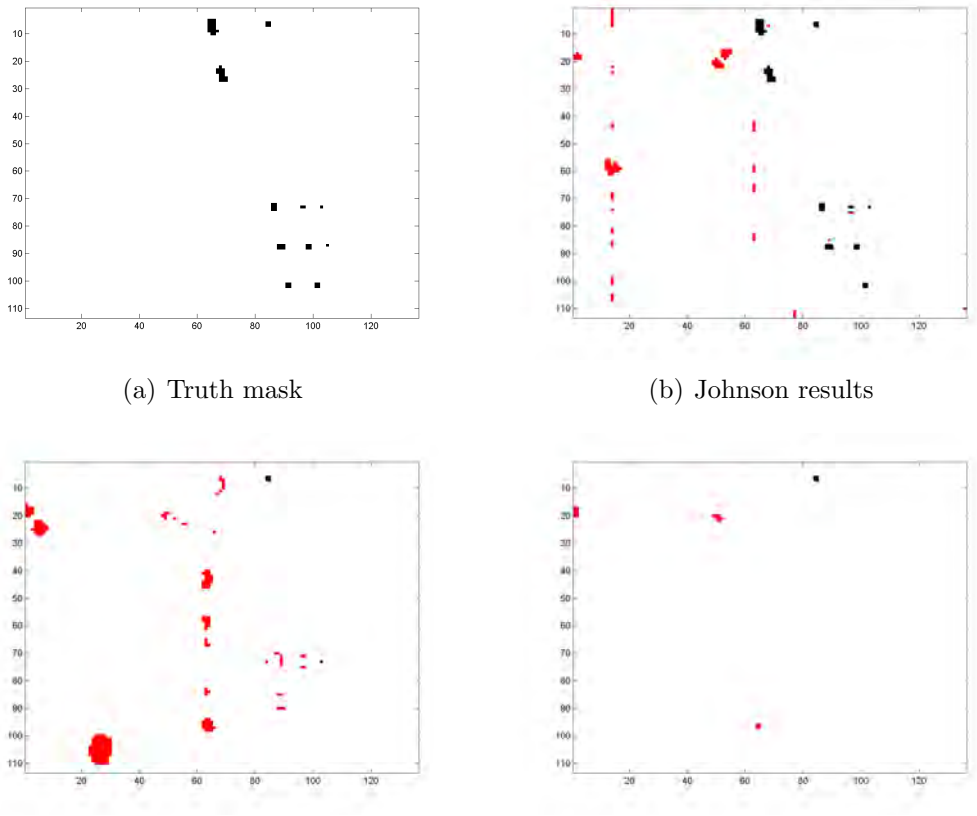
(c) $y^{(1)}$ model results

(d) $y^{(2)}$ model results

Figure 3.14. ARES4F lower half AutoGAD results.

The $y^{(2)}$ model settings were more selective in identifying anomalous pixels. As such, there were a few instances in which the overall false positives were greatly reduced but at the cost of true positives. Figure 3.15 depicts an example in which this occurred on image 3F. In this image, the $y^{(2)}$ model correctly identifies one anomalous cluster while the $y^{(1)}$ model correctly identifies 2 clusters and the Johnson settings label nine anomalous clusters. There is a clear improvement from the $y^{(1)}$ to the $y^{(2)}$ model. The settings provided by Johnson yield better results. However, a one-to-one comparison between Johnson's settings and the RPD model settings from $y^{(1)}$ or $y^{(2)}$ do not provide a fair assessment as Johnson's settings were trained on the

entire set of images. The false alarm rate for the Johnson settings and the $y^{(1)}$ model are drastically larger than the false alarm rate from the $y^{(2)}$ model. This example shows the utility of considering both true positives, engineering solution, and label accuracy, user viewpoint. Ignoring the importance of both viewpoints results in a large number of image chips containing only background, no man-made objects, for the analyst to assess.



(a) Truth mask

(b) Johnson results

(c) $y^{(1)}$ model results

(d) $y^{(2)}$ model results

Figure 3.15. ARES3F lower half AutoGAD results.

While individual image results varied on average, the settings identified by the $y^{(2)}$ model provided more accurate results with only a slight difference in TPF. In practice, an analyst would spend less time checking false anomalies and would be able to process more images. Additional information was gleaned from a validation experiment four images free of true anomalous clusters. The anomaly free validation images were

considered due to the assumption that true anomalies are sparse and most images will contain no true anomalies. Average FPFs of 0.001 and 0.003 were observed for the $y^{(1)}$ model and Johnson settings respectively. Increased label accuracy achieved by the $y^{(2)}$ model resulted in an average FPF of only 0.0007.

Ten more validation images were considered that were collected from various altitudes higher than the ones used to train the $y^{(1)}$ and $y^{(2)}$ models. Table 3.9 gives the average TPF, FPF and LA for the training, test and two validation sets for the $y^{(1)}$, $y^{(2)}$ and Johnson settings respectively. The results from the higher altitude test images suggest, unfortunately, that robust settings need to be based on sensor altitude. However, it is of interest to note that the $y^{(2)}$ model performs better than the $y^{(1)}$ model on both sets of validation images. Additionally, the $y^{(2)}$ model yielded lower FPF than Johnson's settings in the validation images.

Table 3.9. Average results for $y^{(1)}$, $y^{(2)}$ and Johnson settings.

	Model	TPF	FPF	LA
Train	$y^{(1)}$	0.62	0.0100	0.40
	$y^{(2)}$	0.67	0.0037	0.65
	Johnson	0.66	0.0100	0.46
Test	$y^{(1)}$	0.74	0.0074	0.49
	$y^{(2)}$	0.66	0.0042	0.58
	Johnson	0.70	0.0100	0.45
Val - high altitude	$y^{(1)}$	0.30	0.0184	0.17
	$y^{(2)}$	0.46	0.0096	0.35
	Johnson	0.56	0.0134	0.33
Val - no anomalies	$y^{(1)}$	N/A	0.0013	N/A
	$y^{(2)}$	N/A	0.0007	N/A
	Johnson	N/A	0.0032	N/A

3.4 Conclusions

In this chapter, we derived the expected value and variance models for RPD with $N \times N$ considered. This higher order model, $y^{(2)}$, was then used to identify optimal settings for AutoGAD across various HYDICE images. The settings found from the

$N \times N$ model improved label accuracy and false alarm rate while maintaining a consistent true positive rate as compared with the optimal settings found using a standard RPD model, $y^{(1)}$. Further inspection of clusters of falsely identified anomalous pixels (FP) also revealed a reduction in the average number of falsely identified image chips requiring a second look from an analyst or the cueing of a sensor, whether aerial or space-borne, to gain further insight on the area of interest when including $N \times N$ terms.

IV. Concluding Remarks

This dissertation presents RPD concepts related to HSI anomaly detection algorithms. The research areas described are broad enough that applications beyond the realm of HSI can incorporate the ideas and achieve significant improvements. The chapters in this document provide a methodology for implementing each concept.

4.1 Original Contributions

Chapter 2 describes a method for selecting hyperspectral image training and test subsets from a small sample size yielding consistent RPD results based on three noise features: Fisher ratio, percent targets and number of clusters. These subsets are not necessarily orthogonal, but still provide improvements over random training and test subset assignments by maximizing the volume and average distance between image noise characteristics of their respective sets. The small sample training and test selection (SSTATS) method was contrasted with randomly selected training sets as well as training sets chosen from the CADEX and DUPLEX algorithms through the use of simulations and an application involving the RX anomaly detector. When considering training sets from a small sample size, if model validation beyond the range of variables in the training set is a concern, the SSTATS algorithm provides superior performance in contrast with CADEX, DUPLEX or randomly selected training sets.

Chapter 3 removes the standard RPD assumption that squared noise terms and noise by noise interactions are negligible by deriving the mean and variance models for RPD with $N \times N$ considered. This higher order model, $y^{(2)}$, was then used to identify optimal settings for AutoGAD across various HYDICE images. The settings found from the $N \times N$ model improved label accuracy and false alarm rate while maintaining a consistent true positive rate as compared with the optimal settings found using a standard RSM model, $y^{(1)}$. A significant reduction in the average number of falsely

identified image chips was observed by an inspection of clusters of falsely identified anomalous pixels (FP) when including $N \times N$ terms. This reduction leads to fewer images requiring a second look from an analyst or the cueing of a sensor, whether aerial or space-borne, to gain further insight on the area of interest.

4.2 Suggested Future Work

In the course of studying RPD concepts, some potential extensions to this research became apparent. Some potential extensions to this research include:

- Consider new subset size rather than $N = \frac{n}{2}$. It was assumed that training and test subsets would contain the same number of elements or images. CADEX and DUPLEX are both capable of creating training and test sets with varied sizes. SSTATS could easily be adapted to have unequal training and test subsets. A study is suggested to assess the predictive power of models based on varied sizes of training and test sets to identify the optimal training to test set ratio currently assumed as 1:1.

- Expand SSTATS to include across subset measures. SSTATS currently creates training and test subsets based solely on within measures, measures that consider volume or spacing within a given subset. Adding a measure to assess the difference across the training and test subsets could provide improved results.

- Use SSTATS to compare algorithm performance. This research laid the framework to identify optimal settings for anomaly detector algorithms allowing a comparison to select the "best" algorithm. The next logical step would be to actually compare anomaly detector algorithm performance when both algorithms are trained from the same training set.

Bibliography

[1] Adler-Golden, S.M. et al. "Remote bathymetry of the littoral zone from AVIRIS, LASH, and QuickBird imagery," *Geoscience and Remote Sensing, IEEE Transactions on*, *43*(2):337 – 347 (feb. 2005).

[2] Anderson, T.W. *An introduction to multivariate statistical analysis*. Wiley publications in statistics, Wiley, 1958.

[3] Anderson, V.L. and R.A. McLean. *Design of experiments: a realistic approach*. Statistics, textbooks and monographs, M. Dekker, 1974.

[4] Ashton, E.A. "Detection of subpixel anomalies in multispectral infrared imagery using an adaptive Bayesian classifier," *Geoscience and Remote Sensing, IEEE Transactions on*, *36*(2):506 –517 (mar 1998).

[5] Bachmann, C.M. et al. "Automatic classification of land cover on Smith Island, VA, using HyMAP imagery," *Geoscience and Remote Sensing, IEEE Transactions on*, *40*(10):2313 – 2330 (oct 2002).

[6] Banerjee, A., et al. "Fast Hyperspectral Anomaly Detection via SVDD." *Image Processing, 2007. ICIP 2007. IEEE International Conference on4*. IV –101 –IV –104. 16 2007-oct. 19 2007.

[7] Banerjee, A. et al. "A support vector method for anomaly detection in hyperspectral imagery," *Geoscience and Remote Sensing, IEEE Transactions on*, *44*(8):2282 –2291 (2006).

[8] Benediktsson, J.A., et al. "Classification of hyperspectral data from urban areas based on extended morphological profiles," *Geoscience and Remote Sensing, IEEE Transactions on*, *43*(3):480 – 491 (march 2005).

[9] Berk, A., et al. *MODTRAN: A moderate resolution model for LOWTRAN 7*. tech. report GL-TR-89-0122, Geophysics lab, Bedford, MA, 1989.

[10] Borror, C.M., et al. "Evaluation of statistical designs for experiments involving noise variables," *Journal of Quality Technology*, *34*(1):54–70 (2002).

[11] Brando, V.E. and A.G. Dekker. "Satellite hyperspectral remote sensing for estimating estuarine and coastal water quality," *Geoscience and Remote Sensing, IEEE Transactions on*, *41*(6):1378 – 1387 (june 2003).

[12] Carlotto, M.J. "A cluster-based approach for detecting man-made objects and changes in imagery," *Geoscience and Remote Sensing, IEEE Transactions on*, *43*(2):374 – 387 (feb. 2005).

[13] Chang, C.I. "Orthogonal subspace projection (OSP) revisited: a comprehensive study and analysis," *Geoscience and Remote Sensing, IEEE Transactions on*, *43*(3):502 – 518 (Mar 2005).

[14] Chang, C.I. and S. Chiang. "Anomaly detection and classification for hyperspectral imagery," *Geoscience and Remote Sensing, IEEE Transactions on*, *40*(6):1314 – 1325 (Jun 2002).

[15] Chang, C.I. and Q. Du. "Estimation of number of spectrally distinct signal sources in hyperspectral imagery," *42*(3):608–619 (March 2004).

[16] Chang, C.I. and H. Ren. "An experiment-based quantitative and comparative analysis of target detection and image classification algorithms for hyperspectral imagery," *Geoscience and Remote Sensing, IEEE Transactions on*, *38*(2):1044 –1063 (mar 2000).

[17] Chiang, Shao-Shan and C.I. Chang. "Discrimination measures for target classification." *Geoscience and Remote Sensing Symposium, 2001. IGARSS '01. IEEE 2001 International4*. 1871 –1873 vol.4. 2001.

[18] Chiang, S.S et al. "Unsupervised Hyperspectral Image Analysis Using Independent Component Analysis.".

[19] Copeland, K.A.F. and P.R. Nelson. "Dual response optimization via direct function minimization," *Journal of Quality Technology*, *28*:331–336 (1996).

[20] Daughtry, C.S.T. and C.L. Walthall. "Spectral Discrimination of Cannabis sativa L. Leaves and Canopies," *Remote Sensing of Environment*, *64*(2):192 – 201 (1998).

[21] Davis, C.O. et al. "Using hyperspectral imaging to characterize the coastal environment." *Aerospace Conference Proceedings, 2002. IEEE3*. 3–1515 – 3–1521 vol.3. 2002.

[22] Davis, M. *Using multiple robust parameter design techniques to improve hyperspectral anomaly detection algorithm performance*. MS Thesis, AFIT/GOR/ENS/09-05, Air Force Institute of Technology (AU), Wright-Patterson AFB, OH, March 2009.

[23] Del Castillo, E. and D.C. Montgomery. "A nonlinear programming solution to the dual response problem," *Journal of Quality Technology*, *25*:199–204 (1993).

[24] Dillon, W.R. and M. Goldstein. *Multivariate analysis methods and applications*. New York: John Wiley and Sons Inc., 1984.

[25] Duda, R.O., et al. *Pattern classification*. New York: John Wiley and Sons, Inc., 2001.

[26] Duran, O. and M. Petrou. "A Time-Efficient Method for Anomaly Detection in Hyperspectral Images," *Geoscience and Remote Sensing, IEEE Transactions on*, *45*(12):3894 –3904 (dec. 2007).

[27] Eismann, M.T., et al. "Automated Hyperspectral Cueing for Civilian Search and Rescue," *Proceedings of the IEEE*, *97*(6):1031 –1055 (june 2009).

[28] Elerding, G.T., et al. "Wedge imaging spectrometer: application to drug and pollution law enforcement," *1479*(1):380–392 (1991).

[29] Friend, M.A. and K.W. Bauer. "An Entropy-based Scheme for Automatic Target Recognition," *The Journal of Defense Modeling and Simulation: Applications, Methodology, Technology*, *7*(2):103–114 (2010).

[30] Goldberg, H. et al. "Kernel Eigenspace Separation Transform for Subspace Anomaly Detection in Hyperspectral Imagery," *Geoscience and Remote Sensing Letters, IEEE*, *4*(4):581 –585 (oct. 2007).

[31] Goodenough, D.G. et al. "Processing Hyperion and ALI for forest classification," *Geoscience and Remote Sensing, IEEE Transactions on*, *41*(6):1321 – 1331 (june 2003).

[32] Green, R.O. et al. "In-flight validation and calibration of the spectral and radiometric characteristics of the Airborne Visible/Infrared Imaging Spectrometer (AVIRIS)," (1990).

[33] Haboudane, D. et al. "Hyperspectral vegetation indices and novel algorithms for predicting green LAI of crop canopies: Modeling and validation in the context of precision agriculture," *Remote Sensing of Environment*, *90*(3):337 – 352 (2004).

[34] Harsanyi, J.C. and C.I. Chang. "Hyperspectral image classification and dimensionality reduction: an orthogonal subspace projection approach," *Geoscience and Remote Sensing, IEEE Transactions on*, *32*(4):779 –785 (jul 1994).

[35] Hsueh, M. and C.I. Chang. "Adaptive causal anomaly detection for hyperspectral imagery." *Geoscience and Remote Sensing Symposium, 2004. IGARSS '04. Proceedings. 2004 IEEE International5*. 3222 –3224 vol.5. sept. 2004.

[36] Hyvärinen, A. et al. *Independent component analysis*. Adaptive and learning systems for signal processing, communications, and control, J. Wiley, 2001.

[37] Johnson, R.J. *Improved feature extraction, feature selection and identification techniques that create a fast unsupervised hyperspectral target detection algorithm*. MS Thesis, AFIT/GOR/ENS/08-07, Air Force Institute of Technology (AU), Wright-Patterson AFB, OH, March 2008.

[38] Kelton, W.D., et al. *Simulation with Arena*. New York, NY, USA: McGraw-Hill, Inc., 2004.

[39] Kennard, R. W. and L. A. Stone. "Computer Aided Design of Experiments," *Technometrics*, *11*(1):pp. 137–148 (1969).

[40] Kim, K. and D.K.J. Lin. "Response surface optimization: a fuzzy modeling approach," *Journal of Quality Technology*, *30*:1–10 (1998).

[41] Koksoy, O. "Multiresponse robust design: Mean square error (MSE) criterion," *Applied Mathematics and Computation*, *175*(2):1716 – 1729 (2006).

[42] Koksoy, O. and N. Doganaksoy. "Joint optimization of mean and standard deviation using response surface methods," *Journal of Quality Technology*, *35*:239–252 (2003).

[43] Kruse, F.A. et al. "Comparison of EO-1 Hyperion and airborne hyperspectral remote sensing data for geologic applications." *Aerospace Conference Proceedings, 2002. IEEE3*. 3–1501 – 3–1513 vol.3. 2002.

[44] Kwan, C., et al. "A novel approach for spectral unmixing, classification, and concentration estimation of chemical and biological agents," *Geoscience and Remote Sensing, IEEE Transactions on*, *44*(2):409 – 419 (feb. 2006).

[45] Lam, S.W. and L.C. Tang. "A graphical approach to the dual response robust design problems." *Reliability and Maintainability Symposium, 2005. Proceedings. Annual*. 200 – 206. 24-27, 2005.

[46] Landgrebe, D. and E. Malaret. "Noise in Remote-Sensing Systems: The Effect on Classification Error," *IEEE Transactions on Geoscience and Remote Sensing*, *24*:294–300 (March 1986).

[47] Landgrebe, D.A. "Hyperspectral image data analysis," *Signal Processing Magazine, IEEE*, *19*(1):17–28 (Jan 2002).

[48] Landgrebe, D.A. "Multispectral land sensing: where from, where to?," *Geoscience and Remote Sensing, IEEE Transactions on*, *43*(3):414 – 421 (march 2005).

[49] Landgrebe, D.A. *Signal Theory Methods in Multispectral Remote Sensing*. John Wiley and Sons, Inc., 2005.

[50] Lim, J.S. *Two-dimensional signal and image processing*. Prentice-Hall signal processing series, Prentice Hall, 1990.

[51] Lin, DKJ. and W. Tu. "Dual response surface optimization," *Journal of Quality Technology*, *27*:34–39 (1995).

[52] Liu, W. and C.I. Chang. "Multiple-Window Anomaly Detection for Hyperspectral Imagery." *Geoscience and Remote Sensing Symposium, 2008. IGARSS 2008. IEEE International2*. II–41 –II–44. july 2008.

[53] Loeffelholz, B.J. *Neural extensions to robust parameter design*. Dissertation, Air Force Institute of Technology (AU), Wright-Patterson AFB, OH, September 2010. AFIT/DS/ENS/10-03.

[54] Manolakis, D. and G. Shaw. "Detection algorithms for hyperspectral imaging applications," *IEEE Signal Processing Magazine*, *19*(1):29–43 (Jan 2002).

[55] Mao, K.Z. "RBF neural network center selection based on Fisher ratio class separability measure," *Neural Networks, IEEE Transactions on*, *13*(5):1211 – 1217 (sep 2002).

[56] Miller, M.K. *Exploitation of intra-spectral band correlation for rapid feature selection and target identification in hyperspectral imagery*. MS Thesis, AFIT/GOR/ENS/09-10, Air Force Institute of Technology (AU), Wright-Patterson AFB, OH, March 2009.

[57] Mindrup, F.M., et al. "Modeling noise in a framework to optimize the detection of anomalies in hyperspectral imaging." *Intelligent Engineering Systems through Artificial Neural Networks: Computational Intelligence in Architecting Complex Engineering Systems20*, edited by Cihan H. Dagli. 517–524. ASME, Nov 2010.

[58] Mindrup, F.M., et al. "Selecting training and test images for optimized anomaly detection algorithms in hyperspectral imagery through robust parameter design," *Proc. SPIE 8048*(1):80480C (2011).

[59] Montgomery, D. *Design and analysis of experiments* (7th Edition). Wiley, 2009.

[60] Montgomery, D.C., et al. *Introduction to linear regression analysis* (Third Edition). Wiley Series in Probability and Statistics: Texts, References, and Pocketbooks Section, Wiley-Interscience, New York, 2001.

[61] Mosher, T. and M. Mitchell. "Hyperspectral imager for the coastal ocean (HICO)." *Aerospace Conference, 2004. Proceedings. 2004 IEEE1*. 6 vol. (xvi+4192). march 2004.

[62] Myers, R.H. and W.H. Carter. "Response surface techniques for dual response systems," *Technometrics*, *15*:301–317 (1973).

[63] Myers, R.H., et al. *Response surface methodology* (Third Edition). Wiley Series in Probability and Statistics.

[64] Nair, V.N. "Taguchi's parameter design: A panel discussion," *Technometrics*, *34*:127–161 (1992).

[65] Papadimitriou, C. H. and K. Steiglitz. *Combinatorial optimization: algorithms and complexity*. Mineola, NY: Dover Publications Inc., 1998. Corrected reprint of the 1982 original.

[66] Pelleg, D. and A. Moore. *X-means: Extending K-means with efficient estimation of the number of clusters*. Technical Report, Pittsburgh, PA: Carnegie Mellon University, 2000.

[67] Reed, I. S. and X. Yu. "Adaptive multiple-band CFAR detection of an optical pattern with unknown spectral distribution," *IEEE transactions on acoustics, speech and signal processing*, *38*(10):1760–1770 (Oct 1990).

[68] Renard, N. and S. Bourennane. "Dimensionality Reduction Based on Tensor Modeling for Classification Methods," *Geoscience and Remote Sensing, IEEE Transactions on*, *47*(4):1123 –1131 (april 2009).

[69] Robinson, Timothy J., Connie M. Borror and Raymond H. Myers. "Robust parameter design: A review," *Quality and reliability engineering international*, *20*:81–101 (2004).

[70] Roessner, S., et al. "Automated differentiation of urban surfaces based on airborne hyperspectral imagery," *Geoscience and Remote Sensing, IEEE Transactions on*, *39*(7):1525 –1532 (jul 2001).

[71] Schmidt, S.R and R.G Launsby. *Understanding industrial designed experiments* (3rd ed Edition). Colorado Springs, Colo. : Air Academy Press, 1992.

[72] Searle, S.R. *Linear models*. Wiley classics library, Wiley, 1997.

[73] Shaibu, A. and Byung Cho. "Another view of dual response surface modeling and optimization in robust parameter design," *The International Journal of Advanced Manufacturing Technology*, *41*:631–641 (2009). 10.1007/s00170-008-1509-2.

[74] Shi, M and G. Healey. "Using multiband correlation models for the invariant recognition of 3-D hyperspectral textures," *Geoscience and Remote Sensing, IEEE Transactions on*, *43*(5):1201 – 1209 (may 2005).

[75] Smetek, T.E. *Hyperspectral imagery target detection using improved anomaly detection and signature matching methods*. Dissertation, AFIT/DS/ENS/07-07, Air Force Institute of Technology (AU), Wright-Patterson AFB, OH, June 2007.

[76] Smetek, T.E. and K.W. Bauer. "Finding Hyperspectral Anomalies Using Multivariate Outlier Detection." *Aerospace Conference, 2007 IEEE*. 1 –24. march 2007.

[77] Smith, G.M. and E.J. Milton. "The use of the empirical line method to calibrate remotely sensed data to reflectance," *International Journal of Remote Sensing*, *20*(13):2653–2662 (1999).

[78] Smith, W.F. *Experimental design for formulation*, *15*. ASA-SIAM Series on Statistics and Applied Probability. Philadelphia, PA: Society for Industrial and Applied Mathematics (SIAM), 2005.

[79] Snee, R.D. "Validation of Regression Models: Methods and Examples," *Technometrics*, *19*(4):pp. 415–428 (1977).

[80] Stein, D. "Application of the normal compositional model to the analysis of hyperspectral imagery." *Advances in Techniques for Analysis of Remotely Sensed Data, 2003 IEEE Workshop on*. 44 – 51. oct. 2003.

[81] Stein, D.W.J., et al. "Anomaly detection from hyperspectral imagery," *Signal Processing Magazine, IEEE*, *19*(1):58 –69 (January 2002).

[82] Stellman, C., et al. "Real-time hyperspectral detection and cuing," *Optical Engineering*, *39*(7):1928–1935 (2000).

[83] Stone, J.V. *Independent component analysis: a tutorial introduction*. Bradford Books, MIT Press, 2004.

[84] Taitano, Y.P., et al. "A locally adaptable iterative RX detector," *EURASIP J. Adv. Signal Process*, *2010*:11:1–11:10 (February 2010).

[85] Tang, L.C. and K. Xu. "A unified approach for dual response surface optimization," *Journal of Quality Technology*, *34*:437–447 (2002).

[86] Turnbaugh, M.A. *A hybrid template-based composite classifier system*. PhD Dissertation, Air Force Institute of Technology (AU), Wright-Patterson AFB, OH, February 2009.

[87] Varshney, P.K. and M.K. Arora. *Advanced image processing techniques for remotely sensed hyperspectral data*. Springer, 2004.

[88] Vining, G.G. and R.H. Myers. "Combining Taguchi and response surface philosophies: A dual response approach," *Journal of quality technology*, *22*:38–45 (1990).

[89] Yanfeng, G., et al. "Unmixing Component Analysis for Anomaly Detection in Hyperspectral Imagery." *Image Processing, 2006 IEEE International Conference on*. 965 –968. oct. 2006.

[90] Yanfeng, G., et al. "A Selective Kernel PCA Algorithm for Anomaly Detection in Hyperspectral Imagery." *Acoustics, Speech and Signal Processing, 2006. ICASSP 2006 Proceedings. 2006 IEEE International Conference on2*. II. may 2006.

[91] Zadnik, J. *et al.* "Calibration procedures and measurements for the COMPASS hyperspectral imager," *5425*(1):182–188 (2004).

[92] Zare, A., et al. "Vegetation Mapping for Landmine Detection Using Long-Wave Hyperspectral Imagery," *Geoscience and Remote Sensing, IEEE Transactions on*, *46*(1):172 –178 (jan. 2008).

Appendix A. Computer Network Example Data

Table 1.1. Computer network data.

Obs	CS	PS	Serv	Slots	BLK
1	2	150	120	34	0
2	2	150	120	46	0
3	2	150	240	34	0
4	2	150	240	46	0
5	2	450	120	34	0.016
6	2	450	120	46	0
7	2	450	240	34	0.031
8	2	450	240	46	0.003
9	6	150	120	34	0.012
10	6	150	120	46	0
11	6	450	120	46	0.016
12	6	300	180	40	0
13	4	300	180	40	0
14	4	300	180	52	0
15	4	300	180	40	0.016
16	4	300	180	40	0.005
17	4	300	180	40	0.011
18	4	300	180	40	0.02
19	4	300	180	40	0.017
20	4	300	180	40	0.036
21	4	300	180	40	0.007
22	5	400	210	43	0.07

Continued on next page.

Table 1.2. Computer network data (cont).

Obs	CS	PS	Serv	Slots	BLK
23	5	400	210	43	0.082
24	5	400	150	37	0.067
25	5	400	150	37	0.084
26	5	400	150	43	0.027
27	5	400	210	40	0.106
28	5	400	210	40	0.125
29	4	400	210	37	0.084
30	4	400	210	37	0.101
31	4	400	210	40	0.063
32	4	400	210	40	0.057
33	4	400	180	40	0.035
34	5	400	180	40	0.074
35	5	400	180	40	0.077
36	5	400	180	37	0.11
37	5	400	180	37	0.139
38	4	400	180	37	0.057
39	4	400	180	37	0.065
40	3	400	210	37	0.017
41	3	400	210	43	0.003
42	3	400	150	37	0.003
43	3	400	150	43	0.005

Appendix B. Tables of AutoGAD Image Results

Table 2.1. Image results for $y^{(1)}$.

	Image	Half	Mean TPF	Var TPF	Mean FPF	Var FPF	Mean LA	Var LA	Mean Time (sec)	Var Time	LA + TPF	Num True Clusters ID'd
Train	1D	Upper	1	0	0.014	0.00	0.23	0.00	1.3	0.01	1.23	3/3
	1F	Upper	0.35	0	0.000	0	0.99	0.00	0.4	0.03	1.33	4/5
	2D	Upper	0.92	0.00	0.004	0.00	0.86	0.00	0.5	0.00	1.79	21/23
	2F	Upper	0.59	0.05	0.010	0.00	0.10	0.00	8.6	0.21	0.69	4/18
	3D	Lower	0.04	0.00	0.016	0.00	0.07	0.00	0.4	0.00	0.11	1/2
	3F	Upper	0.87	0.01	0.027	0.00	0.14	0.00	2.1	6.06	1.00	6/8
	3F	Lower	0.75	0.01	0.013	0.00	0.02	0.00	7.1	0.08	0.77	2/12
	4	Lower	0.51	0.00	0	0	1	0	0.4	0.00	1.51	7/12
	5	Upper	0.75	0	0.012	0.00	0.36	0	1.3	0.02	1.11	5/7
	5F	Upper	0.39	0.00	0.004	0.00	0.19	0.00	13.8	1.99	0.58	12/18
Train avg			0.62		0.01		0.40		3.6		1.01	
Test	1D	Lower	0.96	0.00	0.008	0.00	0.31	0.00	1.2	0.00	1.27	3/3
	1F	Lower	0.93	0.00	0	0	1	0	0.5	0.01	1.93	5/7
	2D	Lower	0.97	0.00	0.001	0.00	0.95	0.00	0.7	0.00	1.92	19/23
	2F	Lower	0.86	0.00	0.001	0.00	0.26	0.03	7.0	10.47	1.13	4/12
	3D	Upper	0.39	0.00	0.007	0.00	0.17	0.03	3.3	0.08	0.56	2/2
	4F	Upper	0.48	0.00	0.025	0.00	0.04	0.00	3.3	0.12	0.52	2/14
	4F	Lower	0.48	0.03	0.007	0.00	0.28	0.03	2.0	0.49	0.76	11/17
	4	Upper	0.96	0.00	0.017	0.00	0.44	0.00	0.3	0.00	1.40	5/7
	5	Lower	0.88	0	0.003	0.00	0.77	0.00	1.0	0.00	1.65	7/9
	5F	Lower	0.46	0.00	0.005	0.00	0.63	0.01	16.5	1.40	1.10	17/27
Test avg			0.74		0.007		0.49		3.6		1.22	
Grand avg			0.68		0.009		0.44		3.6			

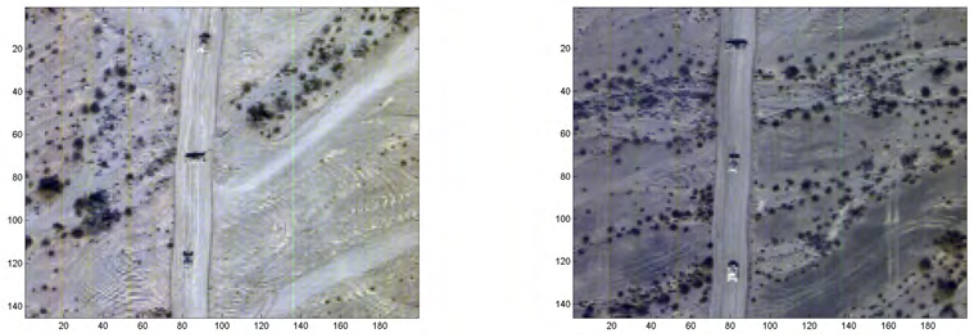
Table 2.2. Image results for $y^{(2)}$.

	Image	Half	Mean TPF	Var TPF	Mean FPF	Var FPF	Mean LA	Var LA	Mean Time (sec)	Var Time	LA + TPF	Num True Clusters ID'd
Train	1D	Upper	0.80	0.00	0.003	0.00	0.51	0.00	1.2	0.01	1.31	3/3
	1F	Upper	0.39	0.00	0	0	1	0	1.0	0.03	1.39	5/5
	2D	Upper	0.92	0.00	0	0	1	0	1.4	2.00	1.92	21/23
	2F	Upper	0.61	0.00	0.002	0.00	0.52	0.11	9.6	0.16	1.13	5/18
	3D	Lower	0.92	0.00	0.007	0.00	0.83	0.00	0.4	0.00	1.75	2/2
	3F	Upper	0.75	0.07	0.019	0.00	0.15	0.00	2.1	5.05	0.90	6/8
	3F	Lower	0.74	0.01	0.001	0.00	0.24	0.03	6.8	0.05	0.98	1/12
	4	Lower	0.52	0.00	0	0	1	0	0.5	0.01	1.52	7/12
	5	Upper	0.76	0.00	0.004	0	0.68	0.00	1.3	0.04	1.45	5/7
	5F	Upper	0.32	0.00	0.001	0.00	0.53	0.01	16.1	1.96	0.85	8/18
Train avg			0.67		0.004		0.65		4.0		1.32	
Test	1D	Lower	0.77	0.00	0.004	0.00	0.41	0.00	1.3	0.02	1.18	3/3
	1F	Lower	0.59	0	0.003	0.00	0.86	0.00	0.8	0.00	1.45	5/7
	2D	Lower	0.95	0.00	0	0	1	0	0.9	0.01	1.95	20/23
	2F	Lower	0.88	0.00	0.001	0.00	0.43	0.07	7.1	9.80	1.31	4/12
	3D	Upper	0.47	0.02	0.007	0.00	0.13	0.00	4.0	0.27	0.59	1/2
	4F	Upper	0.49	0.02	0.010	0.00	0.13	0.00	4.2	0.09	0.63	3/14
	4F	Lower	0.54	0.01	0.001	0.00	0.73	0.03	2.0	0.92	1.27	12/17
	4	Upper	0.68	0.00	0.012	0.00	0.47	0.00	0.8	0.00	1.15	5/7
	5	Lower	0.79	0	0.002	0.00	0.83	0.00	1.5	0.00	1.62	7/9
	5F	Lower	0.46	0.00	0.002	0.00	0.80	0.00	16.8	1.94	1.26	15/27
Test avg			0.66		0.004		0.58		3.9		1.24	
Grand avg			0.67		0.004		0.61		4.0			

Table 2.3. Image results for Johnson's settings.

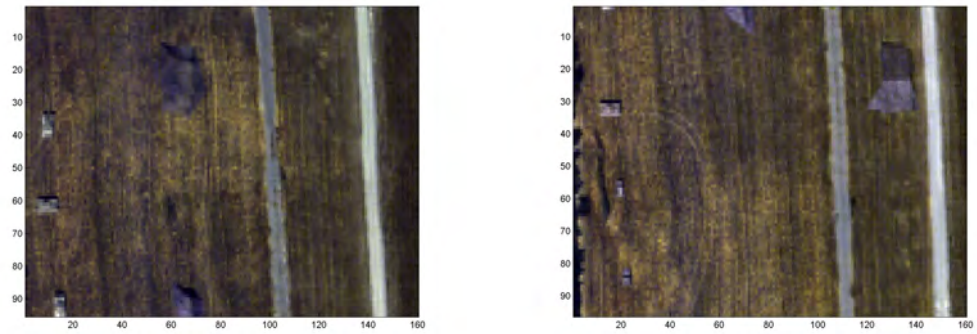
	Image	Half	Mean TPF	Var TPF	Mean FPF	Var FPF	Mean LA	Var LA	Mean Time (sec)	Var Time	LA + TPF	Num TP Clusters ID'd
Train	1D	Upper	0.85	0.01	0.014	0.00	0.21	0	2.5	7.2	1.06	3/3
	1F	Upper	0.98	0	0.001	0.00	0.98	0	0.8	0.1	1.96	5/5
	2D	Upper	0.92	0	0	0.00	1	0	0.5	0	1.92	21/23
	2F	Upper	0.4	0.02	0.007	0.00	0.05	0	8.8	0.1	0.45	17/18
	3D	Lower	0.01	0	0.023	0.00	0.01	0	3.1	1.9	0.02	1/2
	3F	Upper	0.83	0	0.018	0.00	0.19	0	1.3	0	1.02	6/8
	3F	Lower	0.98	0	0.005	0.00	0.44	0.01	1.1	0	1.42	10/12
	4	Lower	0.45	0.01	0	0.00	1	0	1.4	3.7	1.45	8/12
	5	Upper	0.76	0	0.010	0.00	0.45	0	1.3	0	1.21	5/7
	5F	Upper	0.41	0.02	0.004	0.00	0.25	0.02	15.4	1.4	0.66	17/18
Train avg			0.66		0.008		0.46		3.6		1.12	
Test	1D	Lower	0.86	0.01	0.012	0.00	0.16	0	1.6	0.1	1.02	3/3
	1F	Lower	0.93	0	0.007	0.00	0.78	0	1	0	1.71	7/7
	2D	Lower	0.98	0	0.001	0.00	0.95	0	0.6	0	1.93	22/23
	2F	Lower	0.81	0	0.003	0.00	0.13	0.08	9	7.4	0.94	12/12
	3D	Upper	0.35	0.02	0.009	0.00	0.07	0	3.9	1.3	0.42	2/2
	4F	Upper	0.49	0.01	0.016	0.00	0.07	0	2.7	0.6	0.56	9/14
	4F	Lower	0.52	0.02	0.001	0.00	0.71	0.03	1.6	0.4	1.23	13/17
	4	Upper	0.9	0	0.014	0.00	0.49	0	0.4	0	1.39	5/7
	5	Lower	0.83	0.03	0.006	0.00	0.61	0.01	2.3	7.4	1.44	7/9
	5F	Lower	0.47	0	0.003	0.00	0.74	0	17.4	1	1.21	24/27
Test avg			0.71		0.007		0.47		4.1		1.18	
Grand avg			0.69		0.008		0.46		3.8			

Appendix C. Original HYDICE Images



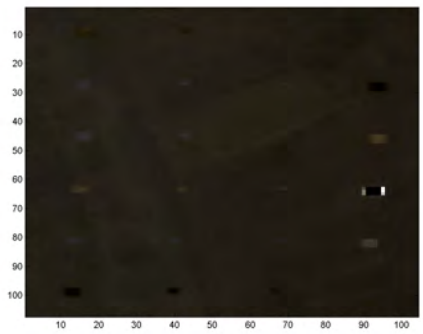
(a) Upper Half (b) Lower Half

Figure 3.1. Image 1D.

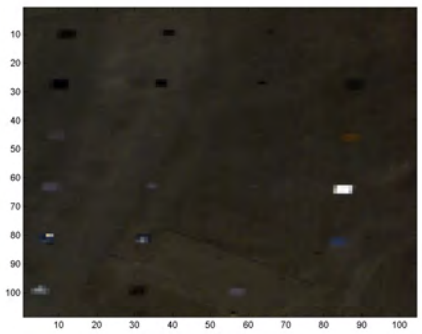


(a) Upper Half (b) Lower Half

Figure 3.2. Image 1F.

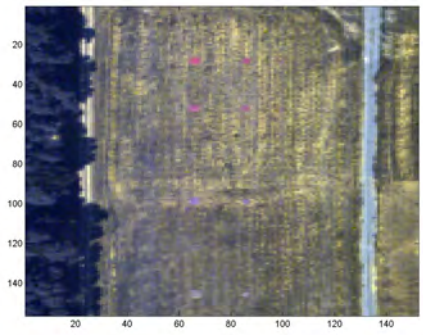


(a) Upper Half

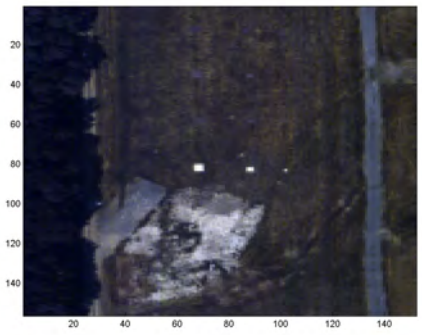


(b) Lower Half

Figure 3.3. Image 2D.



(a) Upper Half



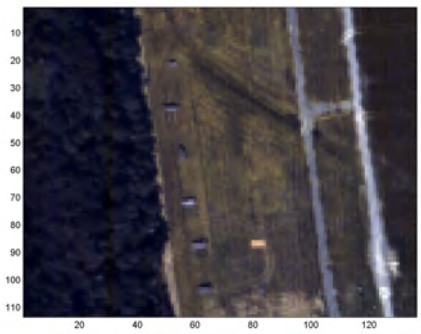
(b) Lower Half

Figure 3.4. Image 2F.

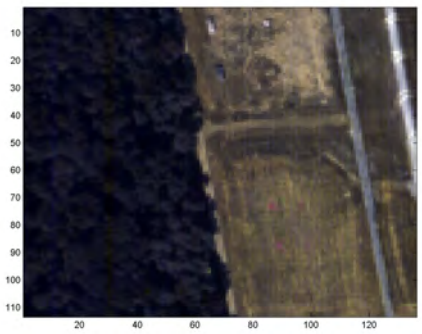
(a) Upper Half

(b) Lower Half

Figure 3.5. Image 3D.

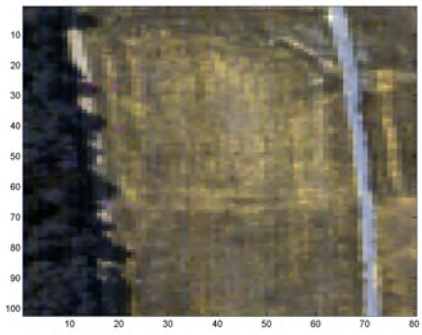


(a) Upper Half

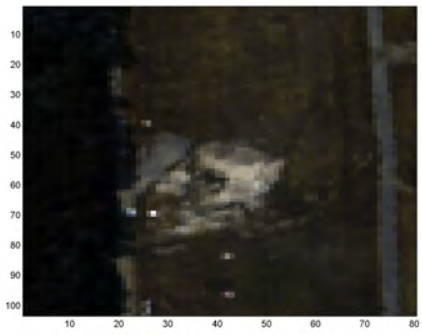


(b) Lower Half

Figure 3.6. Image 3F.

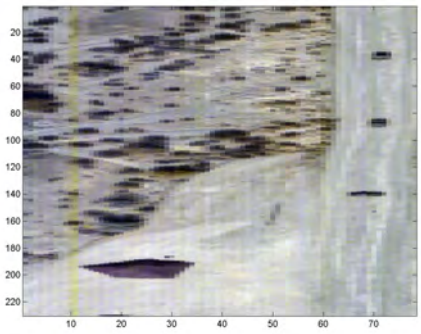


(a) Upper Half

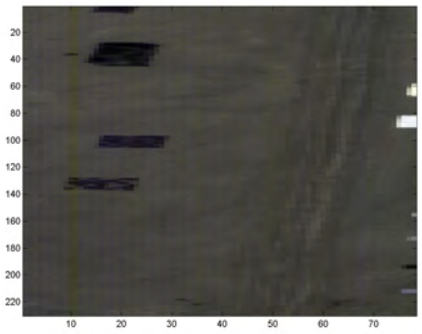


(b) Lower Half

Figure 3.7. Image 4F.

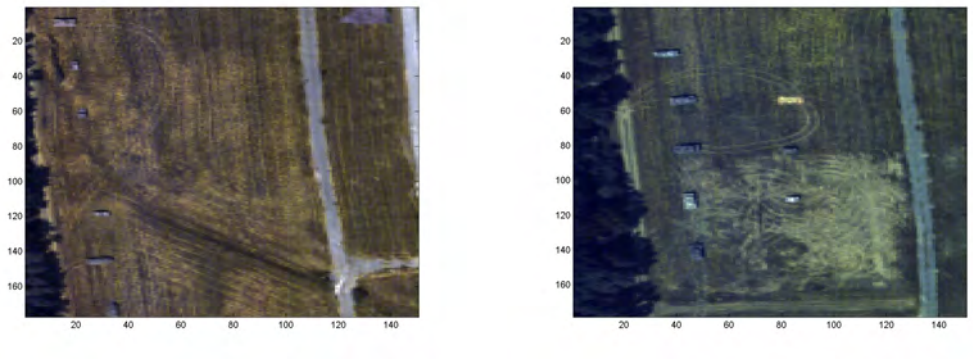


(a) Upper Half



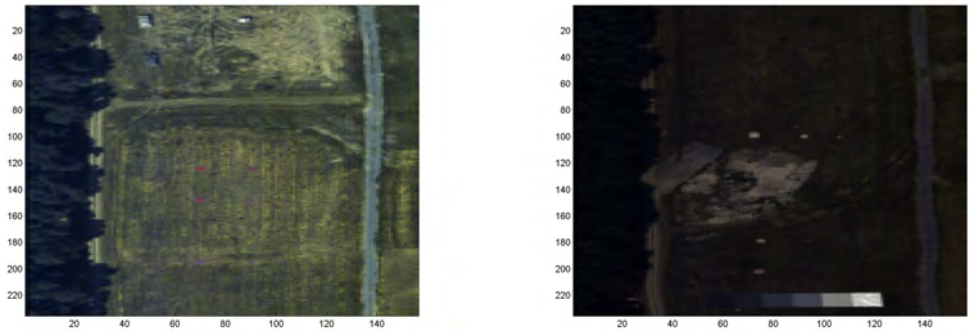
(b) Lower Half

Figure 3.8. Image 4.



(a) Upper Half (b) Lower Half

Figure 3.9. Image 5.



(a) Upper Half (b) Lower Half

Figure 3.10. Image 5F.

Appendix D. Artificial Neural Networks in Engineering (ANNIE) 2010 Conference Paper

MODELING NOISE IN A FRAMEWORK TO OPTIMIZE THE DETECTION OF ANOMALIES IN HYPERSPECTRAL IMAGING

Frank M. Mindrup
Department of Operational Sciences
Air Force Institute of Technology
WPAFB, OH 45433

Trevor J. Bihl
Department of Operational Sciences
Air Force Institute of Technology
WPAFB, OH 45433

Dr. Kenneth W. Bauer Jr.
Department of Operational Sciences
Air Force Institute of Technology
WPAFB, OH 45433

ABSTRACT

Hyperspectral imagery (HSI) has emerged as a valuable tool supporting numerous military and commercial missions. Environmental and other effects diminish HSI classification accuracy. Thus there is a desire to create robust classifiers that perform well in all possible environments. Robust parameter design (RPD) techniques have been applied to determine optimal operating settings. Previous RPD efforts considered an HSI image as categorical noise. This paper presents a novel method utilizing discrete and continuous image characteristics as representations of the noise present. Specifically, the number of clusters, fisher ratio and percent of target pixels were used to generate image training and test sets. Replacing categorical noise with the new image characteristics improves RPD results by correctly accounting for significant terms in the regression model that were otherwise considered categorical factors. Further, traditional RPD assumptions of independent noise variables are invalid for the selected HSI images.

Introduction:

Hyperspectral imagery (HSI) has emerged as a valuable tool supporting numerous military and commercial missions including counter concealment, camouflage and deception, combat search and rescue, counter narcotics, cartography and meteorology to name a few (Manolakis (2002); Landgrebe (2003)). A hyperspectral image, also called an image cube, consists of k spectral bands of an m by n spatial pixel representation of a sensed area. Each pixel in the spectral dimension represents an intensity of energy reflected back to the sensor. All spectral dimensions for a given pixel represent a potential target signature. HSI, by its very nature, can provide a method for identifying at most $(n - 1)$ unique spectral signals, where n is the number of independent bands in an HSI image cube. This is $(n - 1)$ rather than n because one band is used to define the background or noise present in an image. Since HSI contains typically hundreds of bands, this number of signals or targets for classification can be large although bands affected by atmospheric absorption contain little useful information and must be removed and bands that are close to each other are typically correlated.

Davis (2009) describes some pitfalls when performing target classification on hyperspectral images. For instance, the spectral library will most often not contain every possible object, manmade or other to be classified. Some objects may be concealed or disguised to make the spectral signature different from what is contained in the library. In addition, environmental effects such as time of day, relative humidity and imaging angle greatly impact the data reflectance values observed by a sensor. Finally, target prior

probabilities can be very small in comparison to the number of pixels being considered. This leads to a desire to create robust classifiers that perform well in all possible environments or to make very specialized systems that are only used on very specific areas to ensure the spectral library containing spectral signatures of the materials within an image is as accurate and separable as possible.

Typical hyperspectral target detection algorithms can be separated into two classes, anomaly detection and signature matching. Signature matching compares the observed intensities for all bands of an individual pixel with a known spectral signature contained in a library. Anomaly detection compares an individual pixel's mean observed intensity with the mean and variance of the background. Pixels which are statistically different from the background are identified as anomalies.

Previous efforts to develop robust HSI classifiers have utilized robust parameter design (RPD) techniques where each image was considered a categorical noise variable. This paper presents a novel method utilizing discrete and continuous image characteristics as representations of the noise present in an image. Specifically, the number of unique clusters within an image, fisher ratio and percent of target pixels were used to identify image training and test sets. Replacing categorical noise with the new image characteristics improves RPD results by correctly accounting for significant terms in the regression model that were otherwise considered categorical factors. In addition, it is simpler to create models when the noise variables are not categorical.

Robust Parameter Design

Genichi Taguchi proposed an innovative parameter design approach for reducing variation in products and processes in the 1980's. Montgomery (2009) describes RPD as an approach to experimental design that focuses on selecting control factor settings that optimize a selected response while minimizing the variance due to noise factors at that optimum. Control factors are those factors that can be modified in practice while noise factors are often unexplained or uncontrollable in practice. These noise factors can typically be controlled at the research and development level allowing RPD to be performed. There are two methods to model RPD, crossed arrays and combined arrays. This paper will focus on the combined array or response surface method.

RSM methods focus on the roles of control variables on mean and variance in order to provide an estimate at any location of interest. Typically, second-order models are developed when using RSM approaches and higher order interactions are ignored due to the sparcity of effects principle; noise by noise interactions are also assumed to be negligible. A general matrix form of the fitted quadratic response surface model is in the following equation (Myers and Montgomery, 2002)

$$\hat{y}(x,z) = \beta_0 + x'\beta + x'\mathrm{B}x + z'\gamma + x'\Delta z + \varepsilon \qquad (1)$$

where β_0 is the model intercept, β is a vector of the control variable coefficients, B is a matrix of the quadratic control coefficients, γ is a vector of noise variable coefficients, Δ is a matrix of the control by noise interaction coefficients and ε is the pure error of the model which is assumed to be $NID(0,\sigma^2)$. The mean model for the equation can easily be found since the noise variables, z, are assumed to be random variables with $E(z)=0$ and $\mathrm{var}(z) = \sigma_z^2$; further, the noise variables are considered coded random variables centered at zero with limits $\pm a$ representing high and low settings for a particular noise

variable and $\text{cov}(z_i, z_j) = 0, \ \forall\ i \neq j$. Thus the general form of the mean model only includes the control variables and is shown in Montgomery (2009) to be

$$E[\hat{y}(x,z)] = \beta_0 + x'\beta + x'Bx \tag{2}$$

Likewise, the variance model can be found by treating z as a random variable and applying the variance operator to the equation above. The variance model becomes

$$\text{var}[\hat{y}(x,z)] = (\gamma + \Delta' x)'\sigma_z^2(\gamma + \Delta' x) + \sigma^2 \tag{3}$$

where σ^2 is the Mean Square Error found from performing a regression on the design and σ_z^2 is the variance-covariance matrix of z typically assumed to be 1 since the variables are coded. (Myers and Montgomery, 2002)

Categorical Noise

Brenneman and Myers (2003) developed a methodology for treating noise variables categorically for some situations such as when considering different suppliers or brands of equipment as noise. They assert that fewer assumptions are required when considering noise as a categorical variable. Multiple continuous noise variables can be combined into a single categorical noise variable with $r_z + 1$ categories where $P(\text{category } m) = p_m$. It is assumed that these probabilities are known *a priori*. Further, the distribution of this single categorical noise variable is multinomial. The variance-covariance matrix, σ_z^2, from equation (3) becomes

$$\sigma_z^2 = \begin{bmatrix} p_1(1-p_1) & -p_1p_2 & \cdots & -p_1p_{r_z} \\ -p_2p_1 & p_2(1-p_2) & \cdots & -p_2p_{r_z} \\ \vdots & \vdots & \ddots & \vdots \\ -p_{r_z}p_1 & -p_{r_z}p_2 & \cdots & p_{r_z}(1-p_{r_z}) \end{bmatrix}. \tag{4}$$

The prior probabilities required to characterize the variance-covariance matrix might not be available in all situations. Brenneman and Myers also recognized it is possible for robust control settings to be dependent on the p_m. Moreover, the true noise in a hyperspectral image is better characterized by the observable features within an image. Thus, the proposed noise methodology was developed.

Image Noise Methodology

There are several potential observable noise characteristics within an image. This paper will focus on three characteristics: Fisher score, percent of target pixels and

number of clusters. These are not the only characteristics but rather a subset that can be easily calculated within a training set with truth information. Fishers ratio is defined by Lohninger (1999) as a measure for the discriminating power of a variable

$$f = \frac{\mu_1^2 - \mu_2^2}{\sigma_1 + \sigma_2} \tag{5}$$

where μ_1 and σ_1 are the mean and variance of the target class and μ_2 and σ_2 are the mean and variance of the background both defined in a truth matrix. The percent of target pixels can be calculated, if there is a truth map, for each image under test defined as

$$t_i = \frac{v_i}{w_i} \tag{6}$$

where v_i and w_i represent the number of target pixels and background pixels in image i respectively. Clustering was performed using Williams (2007) Matlab ® code of X-means as described by Pelleg and Moore (2000).

Unfortunately, these observed noise characteristics do not fit into the traditional experimental designs since the observations are typically correlated and not orthogonal. Figure 1 shows a classical 2^2 factorial design in circles and an example of an observed set of design points in triangles.

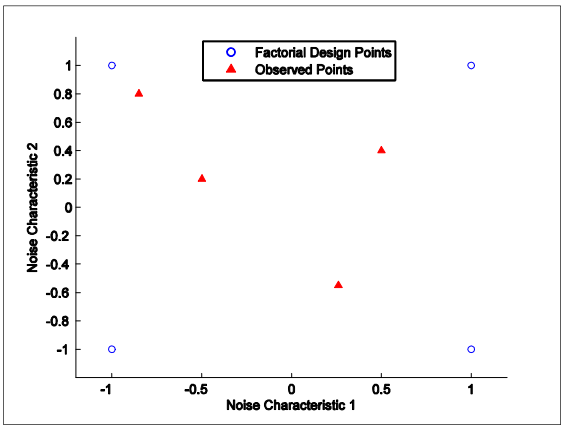


Figure 1: Factorial design (circles) versus observed design (triangles)

The decision for which images to include in the training and test sets is not trivial. Identifying a training and test set of images can be considered a combinatorial optimization problem. Formally, a combinatorial optimization problem is defined as a pair (Ω, f) where Ω is the set of feasible solutions consisting of all possible

combinations of images and f is the cost function (Hall, 2009). Now let R_j be the range for noise factor $j = 1, 2, 3, \ldots n_0$ within a given set of images, ω. Further, let the cost function be defined for the previously defined noise variables (1-fisher's score, 2-percent target, 3-number of clusters) as

$$f_\omega = R_1 + R_2 + R_3 \tag{7}$$

summing the total range across all three noise variables for a given set of images, ω. Let d_i be an indicator variable for image *i* such that

$$d_i = \begin{cases} 1 & \text{if } d_i \text{ is in the training set} \\ 0 & \text{otherwise} \end{cases} \tag{8}$$

The combinatorial optimization problem can thus be solved as a binary integer program

$$\begin{aligned} &\max_{\omega} \; f_\omega = R_1 + R_2 + R_3 \\ &\text{ST.} \; \sum_{i=1}^{n} d_i = k \end{aligned} \tag{9}$$

where *k* is the number of images to be used in training and *n* is the total number of images available. This formulation can result in multiple alternate optimal training sets since some images have extreme values of all noise characteristics. Thus, another binary integer program can be solved on the set of alternative optimals to choose a test set of images. Let $\bar{\omega}$ be the complement of ω for optimal training sets. Assuming all images are to be used in either the training or test set, $\bar{\omega}$ is the set of test images to go with a selected set of training images ω. Let g_i be the indicator variable for image *i* in the test set and *m* be the number of images to include in the test set. Then equation (9) can be adapted to solve for the optimal set of training and test images.

$$\begin{aligned} &\max_{\bar{\omega}} \; f_{\bar{\omega}} = R_1 + R_2 + R_3 \\ &\text{ST.} \; \sum_{i=1}^{n} g_i = m \end{aligned} \tag{10}$$

Results

Training and test images were selected from toy data sets as well as eight images from the Hyperspectral Digital Imagery Experiment (HYDICE). Each image was halved to double the total number of images to 16; image 1 became image halves 1 and 2,

image 2 became image halves 3 and 4 and so on. The observed noise values for all 16 image halves are in Table 1.

The algorithm selected images {1,2,4,6,7,8,10,15} for training and {3,5,9,11,12,13,14,16} for test. Note this puts both halves of images one and four in training and six and seven in test. Figure 2 shows a pair wise comparison of the training and test image noise characteristics. There is typically a training and test image near each observed extreme of the chart. This provides adequate separation of the noise characteristics to perform RPD.

Table 1: Observed image noise characteristics

Image	Image Size	Fisher	% Target	# Clusters
1	28855	1.12	0.0298	3
2	29054	1.11	0.0177	8
3	11128	2.74	0.0077	3
4	11232	2.74	0.0086	3
5	10908	1.10	0.0795	10
6	11016	1.10	0.0803	10
7	12276	1.04	0.0506	9
8	12276	1.04	0.0498	9
9	15200	1.12	0.0002	4
10	15360	1.12	0.0004	10
11	23712	2.35	0.0366	8
12	23712	2.09	0.0446	8
13	15368	1.10	0.0637	7
14	15368	1.10	0.0564	10
15	8160	1.07	0.0000	9
16	8240	1.05	0.0015	10

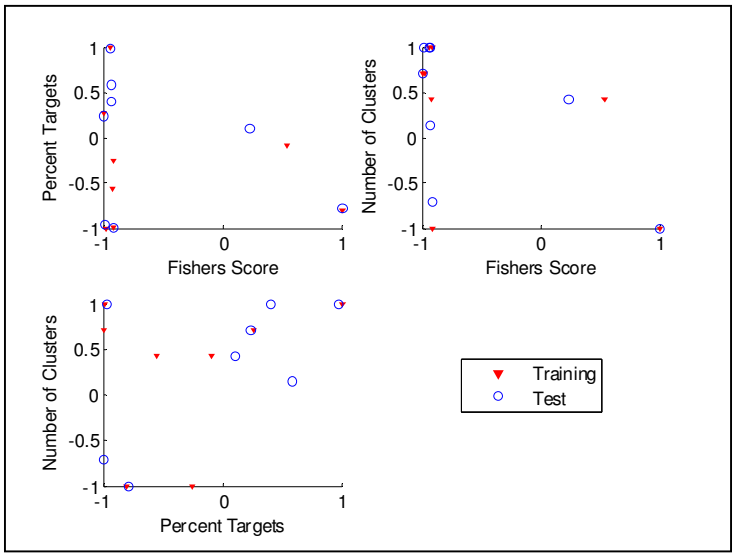


Figure 2: Training and test image pair wise noise characteristics

These images were then used in an RPD of the Autonomous Global Anomaly Detector (AutoGAD) (see Johnson (2008) for the specifics of this algorithm). The experimental design mirrored previous RPD work by Davis (2009) and Miller (2009) using D-optimal designs from Design Expert (see Davis (2009) for specific design information). However, a one-to-one comparison of results will not be presented as Davis and Miller did not halve the images, but rather trained and tested on the complete set of images. Fitting a regression model to the results from training yielded some interesting analysis of variance results. The assumption that no noise by noise interaction exists was not true. R-squared values were as low as 0.5 without noise by noise interactions and were improved by as much as 0.26 when the interactions were included. Table 2 compares the R-squared values with and without noise by noise interactions on four AutoGAD outputs: time, true positive fraction (TPF), false positive fraction (FPF) and target fraction percent (TFP). True positive fraction compares the number of correctly identified pixels with the total number of actual target pixels; false positive fraction compares the total number of falsely labeled (labeled as targets when they were actually noise) pixels with the total number of background pixels. Finally, target fraction percent measures AutoGAD's performance on target clusters. If AutoGAD correctly identifies at least one pixel of a true target cluster, it is counted as a success. TFP is the ratio of these successes to the total number of true target clusters.

Table 2: R-squared values for AutoGAD regression models

Measure	R-squared without interactions	R-squared with interactions
Time	0.5071	0.7719
TPF	0.7145	0.8947
FPF	0.7596	0.8498
TPT	0.5092	0.7527

Conclusions and Future Work

In this paper, we developed a heuristic to identify training and test sets of hyperspectral images for use in RPD based on three continuous noise characteristics of the images. The training and test sets were shown to provide excellent separation of observed noise characteristics. The heuristic was applied to eight images for anomaly detection by AutoGAD.

Future research will include a new mean and variance model with noise by noise interactions to generate a more adequate regression model for RPD. This model will be compared with a neural network representation. Also, D-optimal designs will be compared with the image noise methodology proposed to assess performance characteristics such as time to generate a set of training and test images as well as the separation of the images within each set.

References

Brenneman, William and William Myers. Robust Parameter Design with Categorical Noise Variables," Journal of Quality Technology, 35 (4):335-341 (2003).

Davis, Matthew. Using multiple robust parameter design techniques to improve hyperspectral anomaly detection algorithm performance. Ms thesis, Air Force Institute of Technology (AU), Wright-Patterson AFB, OH, March 2009. AFIT/GOR/ENS/09-05.

Hall, Shane N. OPER 623 approximation and heuristic search methods. Course notes, September 2009.

Johnson, Robert J. Improved feature extraction, feature selection and identification techniques that create a fast unsupervised hyperspectral target detection algorithm. Ms thesis, Air Force Institute of Technology (AU), Wright-Patterson AFB, OH, March 2008. AFIT/GOR/ENS/08-07.

Landgrebe, David A. Signal Theory Methods in Multispectral Remote Sensing. Wiley. Hoboken, NJ 2003.

Lohninger, H.. Teach/Me data analysis. Springer-Verlag, Berlin-New York-Tokyo, 1999.

Manolakis, D. Detection algorithms for hyperspectral imaging applications. Report ESCTR-2001-044, Lincoln Laboratory: Massachusetts Institute of Technology, Lexington, Massachusetts, Feb 2002.

Miller, Michael K. Exploitation of intra-spectral band correlation for rapid feature selection and target identification in hyperspectral imagery. MS thesis, Air Force Institute of Technology (AU), Wright-Patterson AFB, OH, March 2009. AFIT/GOR/ENS/09-10.

Montgomery, Douglas. Design and analysis of experiments. Wiley, 7th edition, 2009.

Myers, Raymond and D. Montgomery. Response surface methodology. Process and product optimization using designed experiments. Wiley, 2nd edition, 2002.

Pelleg, Dan and A. Moore. X-means: Extending k-means with efficient estimation of the number of clusters. Technical report, Carnegie Mellon University, Pittsburgh, PA, 2000.

Smetek, Timothy E. Hyperspectral imagery target detection using improved anomaly detection and signature matching methods. Dissertation, Air Force Institute of Technology (AU),Wright-Patterson AFB, OH, June 2007. AFIT/DS/ENS/07-07.

Williams, Jason P. Robustness of multiple clustering algorithms on hyperspectral images. MS thesis, Air Force Institute of Technology (AU), Wright-Patterson AFB, OH, March 2007. AFIT/GOR/ENS/07-27.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704–0188

1. REPORT DATE *(DD–MM–YYYY)*	2. REPORT TYPE	3. DATES COVERED *(From — To)*
26–10–2011	Doctoral Dissertation	September 2008-October 2011

4. TITLE AND SUBTITLE

Optimizing hyperspectral imagery anomaly detection algorithms through improved robust parameter design techniques

5a. CONTRACT NUMBER

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)

Mindrup, Francis M., Major, USAF

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Air Force Institute of Technology
Graduate School of Engineering and Management (AFIT/EN)
2950 Hobson Way
WPAFB OH 45433-7765 DSN: 785-3636

8. PERFORMING ORGANIZATION REPORT NUMBER

AFIT/DS/ENS/11-04

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)

Intentionally left blank

10. SPONSOR/MONITOR'S ACRONYM(S)

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION / AVAILABILITY STATEMENT

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED

13. SUPPLEMENTARY NOTES

14. ABSTRACT

Anomaly detection algorithms for hyperspectral imagery (HSI) are an important first step in the analysis chain which can reduce the overall amount of data to be processed. The actual amount of data reduced depends greatly on the accuracy of the anomaly detection algorithm implemented. Most, if not all, anomaly detection algorithms require a user to identify some initial parameters. These parameters (or controls) affect overall algorithm performance. Regardless of the anomaly detector being utilized, algorithm performance is often negatively impacted by uncontrollable noise factors which introduce additional variance into the process. In the case of HSI, the noise variables are embedded in the image under consideration. Robust parameter design (RPD) offers a method to model the controls as well as the noise variables and identify robust parameters. This research identifies image noise characteristics necessary to perform RPD on HSI. Additionally, a small sample training and test algorithm is presented. Finally, the standard RPD model is extended to consider higher order noise coefficients. Mean and variance RPD models are optimized in a dual response function suggested by Lin and Tu. Results are presented from simulations and two anomaly detection algorithms, the Reed-Xiaoli anomaly detector and the autonomous global anomaly detector.

15. SUBJECT TERMS

Robust Parameter Design, Hyperspectral Imagery, Anomaly Detection

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			Dr. Kenneth W. Bauer (ENS)
U	U	U	UU	146	19b. TELEPHONE NUMBER *(include area code)* (937)255-3636x4328; e-mail: Kenneth.Bauer@afit.edu